ABSTRACT

Title of dissertation: IMAGE RETRIEVAL BASED ON COMPLEX DESCRIPTIVE QUERIES

Behjat Siddiquie, Doctor of Philosophy, 2011

Dissertation directed by: Professor Larry S. Davis
Department of Computer Science

The amount of visual data such as images and videos available over web has increased exponentially over the last few years. In order to efficiently organize and exploit these massive collections, a system, apart from being able to answer simple classification based questions such as whether a specific object is present(or absent) in an image, should also be capable of searching images and videos based on more complex descriptive questions. There is also a considerable amount of structure present in the visual world which, if effectively utilized, can help achieve this goal. To this end, we first present an approach for image ranking and retrieval based on queries consisting of multiple semantic attributes. We further show that there are significant correlations present between these attributes and accounting for them can lead to superior performance. Next, we extend this by proposing an image retrieval framework for descriptive queries composed of objects categories, semantic attributes and spatial relationships. The proposed framework also includes a unique multi-view hashing technique, which enables query specification in three different modalities - image, sketch and text.

We also demonstrate the effectiveness of leveraging contextual information to reduce the supervision requirements for learning object and scene recognition models. We present an active learning framework to simultaneously learn appearance and contextual models for scene understanding. Within this framework we introduce new kinds of labeling questions that are designed to collect appearance as well as contextual information and which mimic the way in which humans actively learn about their environment. Furthermore we explicitly model the contextual interactions between the regions within an image and select the question which leads to the maximum reduction in the combined entropy of all the regions in the image (image entropy).

IMAGE RETRIEVAL BASED ON COMPLEX DESCRIPTIVE QUERIES

by

Behjat Siddiquie

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2011

Advisory Committee:
Professor Larry S. Davis, Chair/Advisor
Professor John J. Benedetto, Dean's Representative
Professor Ramani Duraiswami
Professor David W. Jacobs
Professor Hal Daumé III

© Copyright by
Behjat Siddiquie
2011

Dedication

to my parents.

Acknowledgments

The completion of this thesis has been made possible due to support and guidance from several people and I would like to take this opportunity to express my gratitude. I apologize if I have inadvertently left out someone. First and foremost is my advisor Prof. Larry Davis, who showed a lot of faith in my abilities and supported me for the entire duration of my stay here. He gave me a tremendous amount of freedom towards pursuing my own ideas. His high level view of things and intuition were instrumental in bringing many of these ideas to fruition and often steered me away from what would have been a dead end. I thank my committee members - Prof. John Benedetto, Prof. Hal Daume, Prof. Ramani Duraiswami and Prof. David Jacobs for taking time from their busy schedules and agreeing to serve on my dissertation committee. I would also like to thank Dr. Yaser Yacoob for his guidance on the football and VIRAT projects.

I would like to thank Dr. Rogerio Feris for mentoring me during my internship at IBM T J Watson. During the internship, I was exposed to image retrieval, which eventually became the focus of my thesis. The internship at IBM under Rogerio's guidance culminated in the work on multi-attribute based image retrieval. I would like to thank Dr. Ajay Divakaran for a fruitful internship at Sarnoff, where I was exposed to challenging real world problems. I am also indebted to him for his guidance and support during my job search. I would also like to thank Prof. Sharat Chandran, my undergraduate advisor, and Prof. Nikos Paragios, with whom I did an internship, both of whose guidance during my undergraduate days inspired me

to pursue a PhD in computer vision.

I would also like to thank my fellow PhD students and labmates, especially Abhinav, Brandyn, Aniruddha and Shiv, with whom I collaborated extensively during different stages of my PhD and I learnt a lot from each of them. I would also like to thank Vlad, Arpit, Fatemeh, Ryan, Hazem, Stephen, Zhuolin, Sameh, Radu, William, Mohamed, Balaji, Abhishek, Ejaz and Sravanthi, with whom I had several interesting discussions inside the lab and several fun times outside.

Finally, I would like to thank my family. I thank the members of my extended family for their support and guidance. I thank my brother and sister for their love and affection. Lastly but most importantly, I would like to thank my parents for instilling in me the importance of systematic and hard work, giving me the love of learning and for supporting me at each and every stage of my life. This thesis is dedicated to them.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Image Retrieval based on Descriptive Queries

The amount of visual data such as images and videos available over web has increased exponentially over the last few years and there is a need for developing techniques that are capable of efficiently organizing, searching and exploiting these massive collections. In order to effectively do so, a system, apart from being able to answer simple classification based questions such as whether a specific object is present(or absent) in an image, should also be capable of searching and organizing images and videos based on more complex descriptive questions. There is also a considerable amount of structure present in the visual world, for example, there are spatial and semantic relationships present between various object classes and several different object categories often share a common set of visual attributes. Exploiting this additional contextual information is crucial to achieve the goal of effectively searching and organizing visual data. To this end, we have developed an image retrieval and ranking approach, which allows for searching image datasets based on queries that comprise multiple semantic attributes, while also taking into account the correlations and dependencies between the attributes, leading to significantly improved performance [117]. We have further extended this work to enable image search and retrieval based on richer and more descriptive queries, consisting

of objects, attributes and relationships, over large scale datasets [116].

1.1.1 Image Ranking and Retrieval based on Multi-Attribute Queries

We have investigated the problem of image ranking and retrieval based on semantic attributes. Consider the problem of ranking/retrieval of images of people according to queries describing the physical traits of a person. For example, one could search for a suspicious person or a missing person in an archive of surveillance video based on a query such as *"young Asian woman wearing sunglasses"*. While previous approaches [78, 88] have looked at this problem, they completely ignore the fact that these attributes are highly correlated. For example, since the above query contains the attribute *"Asian"*, a relevant person is unlikely to have blonde hair, and is more likely to have black hair. Similarly, since one of the constituent attributes is woman, it is easy to discard images containing people with mustaches and beards since they are male specific attributes. Our work exploits such interdependencies between attributes and also leverages the information contained in non-query attributes to improve retrieval based on multi-attribute queries. In image retrieval, the goal is to return the set of images in a database that are relevant to a query. The aim of ranking is similar, but with additional requirement that the images be ordered according to their relevance to the query. For large scale datasets, it is essential for an image search application to rank the images such that the most relevant images are at the top. Hence, we also consider the problem of image ranking to improve the effectiveness of attribute based image search. While learning to rank

has traditionally been treated as a distinct problem within information retrieval, we propose a joint framework for ranking and retrieval based on a structured learning formulation, where learning to rank or retrieve are simply optimizations of the same model according to different performance measures. We also facilitate training, as well as retrieval and ranking, based on queries consisting of multiple-labels by explicitly utilizing the multi-labeled samples present in the training set for the purpose of learning our model. We demonstrate the effectiveness of our approach for the purpose of searching for people based on multi-attribute queries on two different face datasets - LFW [71] and FaceTracer [78]. While searching for images of people involves only a single object class (human faces), we also perform experiments on the PASCAL [70] dataset to show that our approach is general enough to be utilized for attribute based retrieval of images containing multiple object classes.

1.1.2 Multi-view Hashing for Multi-Modal Image Retrieval based on Complex Descriptive Queries

There have been major advances in the field of information retrieval in the last few years. For example, search engines have become extremely adept at extracting and utilizing structured data from unstructured web pages and are even able to answer simple natural language based questions [145]. Similarly, image retrieval has progressed from retrieving images based on single label queries [81, 82] to multi-label queries [132, 133, 78, 117]. While our work on multi-attribute based retrieval [117], was an attempt towards providing users the ability to retrieve images based

on more descriptive queries. We extend it further by facilitating search and retrieval of image databases based on significantly more complex queries consisting of *objects* and *attributes* along with the spatial and comparative *relationships* between the individual objects. As an example, a user could search for images based on a query such as - (find an image where there is a) "*red car to the left of a yellow car which is in front of a gray building*". Clearly, such queries are significantly more expressive than multi-label queries and allow a user to search for more specific images/scenes based on certain characteristics of the scene.

We also investigate the problem of specifying these complex queries through different modalities. In image retrieval, queries are typically specified using an image, a sketch or a textual description and most current image retrieval approaches fall into one of these three categories. We combine these approaches by proposing a joint framework thats allows the queries to be specified in either of these three modalities - i.e. images, sketches and text. However, employing such a framework for multiple query modalities in a large scale setting is very challenging, mainly due to two factors - Firstly, it requires the capability of storing the database images using compact binary codes that are descriptive enough to contain all the semantic information needed to retrieve images relevant to the complex queries. Secondly a large scale scenario necessitates the ability to perform an efficient nearest neighbor search from a query of each modality to the elements in the database. We address these challenges by proposing a novel multi-view hashing approach capable of hashing multiple views(modalities) of the query and the database elements to the same compact binary hash code enabling efficient storage and retrieval.

1.2 Utilizing Contextual Information to reduce Supervision

Machine learning algorithms need substantial amounts of annotated training data for learning good visual models. However, creating large and comprehensively labeled image datasets is an expensive task as it requires a significant amount of human effort. Researchers have sought to overcome these challenges by exploring techniques that reduce the amount of human supervision required. Such methods include unsupervised learning, semi-supervised learning, active learning, transfer learning and effective utilization of weakly labeled data. We have investigated the feasibility of augmenting these approaches by leveraging contextual information to further reduce the amount of annotation required. In particular, we have proposed an active learning approach for learning contextual object recognition models [118]. In contrast to other active learning approaches in vision, that gather only appearance information, we actively acquire both appearance and contextual information. Moreover, we also exploit the contextual information such as the spatial and comparative relationships between pairs of objects to speed up the process of active learning. We have also examined the possibility of utilizing contextual information to perform transfer learning in an unsupervised manner, for view-invariant object recognition, with promising initial results [119].

1.2.1 Modeling Contextual Interactions for Multi-Class Active Learning

Object recognition is one of the most challenging problems in computer vision. While obtaining annotated visual datasets is the prime way to obtain and create visual knowledge-bases, a major concern is the diversity and quantity of the training examples in labeled datasets as they directly impact the performance of most object recognition approaches. Similarly, the performance of context based approaches improves with an increase in size of the training dataset. Due to the difficulty in obtaining a large amount of human labeling, many recent works have proposed using active learning methods to select the images or regions to be labeled by human annotators with the goal of minimizing the manual annotation effort. These approaches typically utilize the classification uncertainty by asking humans to label examples which are hard to classify using classifiers learned from previously labeled data. However, most of the work in active learning for visual recognition has focused on obtaining labeling for binary classification problems, especially where objects occur in isolation.

We propose an active learning framework to simultaneously learn appearance and contextual models for scene understanding tasks [118]. Existing active learning approaches have focused on utilizing classification uncertainty of regions to select the most ambiguous region for labeling. These approaches, however, ignore the contextual interactions between different regions of the image and the fact that knowing the label for one region provides information about the labels of other regions. We

model contextual interactions between image regions and solicit labels for those regions that yield significant reduction in the combined classification uncertainty of all the regions in the image. Therefore, our criterion selects regions which are likely to yield information about the other confusing regions in the image as well. For example, if an object in an image is labeled as "boat", instead of asking a human to annotate the region below it, we might be able to infer its label as "water" since a "boat" is likely to be on "water". We show that, by systematically selecting the regions to be labeled, one can significantly reduce the annotation costs.

Most active learning approaches in vision ask the annotator to label a region(single object) in an image. Apart from this simple labeling question, we propose asking the annotators two new types of questions that are designed to collect appearance as well as contextual information and which mimic the way humans actively learn about their environment. We introduce linguistic questions, where high confidence regions in a scene are used as anchors to pose questions about the uncertain regions in the scene. For example, in an image, the water region which is usually easy to recognize can be utilized as an anchor to ask questions such as "what is on the water?" and the answer to this question would not only provide us with partial appearance information about the objects on "water" in the image, but also contextual information about which object categories obey the semantic relationship "on" with respect to "water". We also introduce Contextual questions, which help in actively learning concepts. For example, our approach might ask the annotator: "What is the relationship between boat and water?", which can help us learn contextual information directly from the annotator. We introduce a novel

entropy based criterion for active selection of labeling questions based on reduction in labeling uncertainty of all the regions in the image. By considering the joint entropy of the image as opposed to the entropy of individual regions, we generate labeling questions which yield information not only about the region whose label is solicited, but about other regions in the image as well.

1.2.2 Unsupervised Transfer Learning for View-Invariant Object Detection

Here we focus on the problem of vehicle detection in urban surveillance environments. Traffic surveillance cameras are becoming increasingly widespread and government agencies seek to use such cameras not just for monitoring traffic but also to search for suspicious vehicles, which requires accurate detection and localization of each vehicle. However, detection and localization of vehicles in surveillance video, which is typically low resolution, is extremely difficult as it requires dealing with view-invariance. Since it not feasible to obtain labeled training data from each surveillance camera to build view-specific models, and view-invariant object detectors are typically very slow, we leverage contextual information such as scene layout and motion patterns to identify training viewpoints (source domains) for performing transfer learning in an unsupervised manner.

Instead of building a view-invariant detector that can model all possible viewpoint deformations, which is extremely hard, we train simple object detectors for a large number of different viewpoints (source domains) which densely span the

viewpoint space that we want to model. Given a new viewpoint(target domain), we exploit scene geometry and vehicular motion patterns to find closely related viewpoints from the source domain where vehicles are expected to occur in poses similar to the target viewpoint. Our dense representation in the viewpoint space ensures that we are guaranteed to find closely related viewpoints in the source domain. We then transfer the knowledge, in the form of learnt object detection models, trained on the selected viewpoints for detecting vehicles in the new viewpoint. Extensive experimental evaluation on a challenging test set, consisting of images collected from fifty different surveillance cameras, demonstrates that our unsupervised approach, based on simple view-specific object detectors, can outperform complex methods that utilize labeled training data from the target domain, both in terms of speed as well as accuracy.

1.3 Efficient Multiple Kernel Learning for Object Recognition

We have also investigated the problem of combining multiple feature channels for the purpose of efficient object recognition. Many existing context based object recognition and scene understanding methods such as [2] use an appearance based recognition method for modeling the object likelihood of each region in an image and a employ a generative model to represent the contextual relationships between the different regions. Hence having an accurate object recognition model is crucial for the performance of such context based recognition systems. Discriminative kernel based methods, such as SVMs, have been shown to be quite effective for

image classification. To use these methods with several feature channels, one needs to combine the base kernels computed from them. Multiple kernel learning is an effective method for combining the base kernels. However, the cost of computing the kernel similarities of a test image with each of the support vectors for all feature channels is extremely high. We propose an alternate method, where training data instances are selected, using AdaBoost, for each of the base kernels. A composite decision function, which can be evaluated by computing kernel similarities with respect to only these chosen instances, is learnt. This method significantly reduces the number of kernel computations required during testing. Experimental results on the benchmark UCI datasets [58], as well as on a challenging painting dataset, are included to demonstrate the effectiveness of our method.

Chapter 2

Image Ranking andRetrieval based on Multi-Attribute Queries

2.1 Introduction

In the past few years, methods that exploit the semantic attributes of objects have attracted significant attention in the computer vision community. The usefulness of these methods has been demonstrated in several different application areas, including object recognition [69, 68, 79] face verification [77] and image search [88, 78].

In this chapter we address the problem of *image ranking and retrieval based on semantic attributes*. Consider the problem of ranking/retrieval of images of people according to queries describing the physical traits of a person, including facial attributes (e.g. hair color, presence of beard or mustache, presence of eyeglasses or sunglasses etc.), body attributes (e.g. color of shirt and pants, striped shirt, long/short sleeves etc.), demographic attributes (e.g. age, race, gender) and even non-visual attributes (e.g. voice type, temperature and odor) which could potentially be obtained from other sensors. There are several applications that naturally fit within this attribute based ranking and retrieval framework. An example is criminal investigation. To locate a suspect, law enforcement agencies typically gather the physical traits of the suspect from eyewitnesses. Based on the description obtained, entire video archives from surveillance cameras are scanned manually for persons

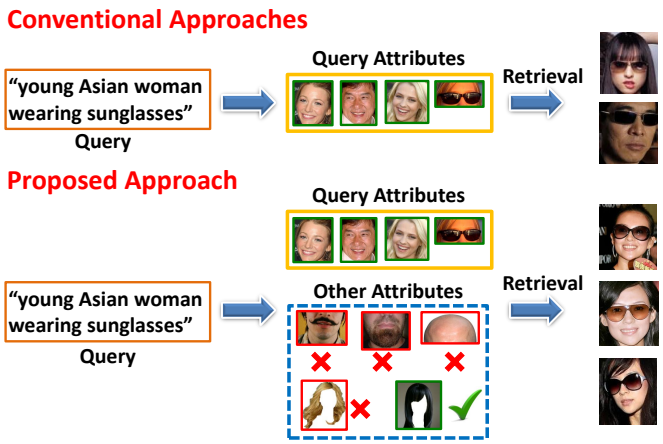


Figure 2.1: Given a multi-attribute query, conventional image retrieval methods such as [88, 78], consider only the attributes that are part of the query, for retrieving relevant images. On the other hand, our proposed approach also takes into account the remaining set of attributes that are not a part of the query. For example, given the query "*young Asian woman wearing sunglasses*", our system infers that relevant images are unlikely to have a *mustache*, *beard* or *blonde hair* and likely to have *black hair*, thereby achieving superior results.

with similar characteristics. This process is time consuming and can be drastically accelerated by an effective image search mechanism.

Searching for images of people based on visual attributes has been previously investigated in [88, 78]. Vaquero et al. [88] proposed a video based surveillance system that supports image retrieval based on attributes. They argue that while face recognition is extremely challenging in surveillance scenarios involving low-resolution imagery, visual attributes can be effective for establishing identities over short periods of time. Kumar et al. have built an image search engine [78] where users can retrieve images of faces based on queries involving multiple visual attributes. How-

ever, these methods do not consider the fact that attributes are highly correlated. For example, a person who has a *mustache* is almost definitely a *male*, or a person who is *Asian* is unlikely to have *blonde hair*.

We present a new framework for multi-attribute image retrieval and ranking, which retrieves images based not only on the words that are part of the query, but also considers the remaining attributes within the vocabulary that could potentially provide information about the query (Figure 2.1). Consider a query such as *"young Asian woman wearing sunglasses"*. Since the query contains the attribute *young*, pictures containing people with gray hair, which usually occurs in older people, can be discounted. Similarly pictures containing bald people or persons with mustaches and beards, which are male specific attributes, can also be discarded, since one of the constituent attributes of the query is *woman*. While an individual detector for the attribute *woman*, will implicitly learn such features, our experiments show that when searching for images based on queries containing fine-grained parts and attributes, explicitly modeling the correlations and relationships between attributes can lead to substantially better results.

In image retrieval, the goal is to return the set of images in a database that are relevant to a query. The aim of ranking is similar, but with additional requirement that the images be ordered according to their relevance to the query. For large scale datasets, it is essential for an image search application to rank the images such that the *most* relevant images are at the top. Ranking based on a single attribute can sometimes seem unnecessary; for example, for a query like "*beard*", one can simply classify images into people with beards and people without beards.

For multi-attribute queries however, depending on the application, one can have multiple levels of relevance. For example, consider a query such as "*man wearing a red shirt and sunglasses*", since sunglasses can be easily removed, it is reasonable to assume that images containing men wearing a red shirt but without sunglasses are also relevant to the query, but perhaps less relevant than images of men with both a red shirt and sunglasses. Hence, we also consider the problem of ranking based on multi-attribute queries to improve the effectiveness of attribute based image search. Instead of treating ranking as a separate problem, we propose a structured learning framework, which integrates ranking and retrieval within the same formulation.

While searching for images of people involves only a single object class (i.e. human faces), we show that our approach is general enough to be utilized for attribute based retrieval of images containing multiple object classes, and outperforms a number of different ranking and retrieval methods on three different datasets - LFW [71] and FaceTracer [78] for human faces and PASCAL [70] for multiple object categories.

There are three key **contributions** of our work: **(1)** We propose a single framework for image ranking and retrieval. Traditionally, *learning to rank* is treated as a distinct problem within information retrieval. In contrast, our approach deals with ranking and retrieval within the same formulation, where learning to rank or retrieve are simply optimizations of the same model according to different performance measures. **(2)** Our approach supports image retrieval and ranking based on multi-label queries. This is non-trivial, as the number of possible multi-label queries for a vocabulary of size L is 2^L. Most image ranking/retrieval approaches deal with this problem by learning separate classifiers for each individual label, and

retrieve multi-label queries by heuristically combining the outputs of the individual labels. In contrast, we introduce a principled framework for training and retrieval of multi-label queries. **(3)** We also demonstrate that attributes within a single object category and even across multiple object categories are interdependent so that modeling the correlations between them leads to significant performance gains in retrieval and ranking.

2.2 Related Work

An approach that has proved extremely successful for document retrieval is *learning to rank* [72, 73, 74, 85], where a ranking function is learnt, given either the pairwise preference relations or relevance levels of the training examples. Similar methods have also been proposed for ranking images, [86]. Several image retrieval methods, which retrieve images relevant to a textual query, adopt a *visual reranking* framework [81, 82, 83, 84], which is a two stage process. In the first stage images are retrieved based purely on textual features like tags(e.g. in Flickr), query terms in webpages and image meta data. The second stage involves reranking or filtering these images using a classifier trained on visual features. A major limitation of these approaches is the requirement of textual annotations for the first stage of retrieval, which are not always available in many applications - for example the surveillance scenario described in the introduction. Another drawback of both the image ranking approaches as well as the *visual reranking* methods is that they learn a separate ranking/classification function corresponding to each query term and hence have to

resort to ad-hoc methods for retrieving/ranking multi-word queries. A few methods have been proposed for dealing with multi-word queries. Notable among them are PAMIR [132] and TagProp [133]. However, these methods do not take into account the dependencies between query terms. We show that there often exist significant dependencies between query words and modeling them can substantially improve ranking and retrieval performance.

Recently, there have been several works which utilize an attribute based representation to improve performance of computer vision tasks. In [69], Farhadi et al. advocate an attribute centric approach for object recognition, and show that in addition to effective object recognition, it helps in describing unknown object categories and detecting unexpected attributes in known object classes. Similarly, Lampert et al. [68] learn models of unknown object categories from attributes based on textual descriptions. Kumar et al. [77] have shown that comparing faces based on facial attributes and other visual traits can significantly improve face verification. Wang and Mori [79] have demonstrated that recognizing attributes and modeling the interdependencies between them can help improve object recognition performance. In general, most of these methods exploit the fact that attributes provide a high level representation which is compact and semantically meaningful.

Tsochantaridis et al. introduced Structured SVMs [26] to address prediction problems involving complex outputs. Structured SVMs provide efficient solutions for structured output problems, while also modeling the interdependencies that are often present in the output spaces of such problems. They have been effectively used for object localization [28] and modeling the cooccurrence relationships be-

tween attributes [79]. The structured learning framework has also been utilized for document ranking [74], which is posed as a structured prediction problem by having the output be a permutation of the documents. In this work, we employ structured learning to pose a single framework for ranking and retrieval, while also modeling the correlations between the attributes.

2.3 Multi Attribute Retrieval and Ranking

We now describe our Multi-Attribute Retrieval and Ranking(MARR) approach. Our image retrieval method is based on the concept of reverse learning. Here, we are given a set of labels $\mathcal{X}$, and a set of training images $\mathcal{Y}$. Corresponding to each label x_i ($x_i \in \mathcal{X}$) a mapping is learned to predict the set of images y ($y \subset \mathcal{Y}$) that contain the label x_i. Since reverse learning has a structured output (set of images) it fits well into the structured prediction framework. Reverse learning was recently proposed in [67], and was shown to be extremely effective for multi-label classification. The main advantage of reverse learning is that it allows for learning based on the minimization of loss functions corresponding to a wide variety of performance measures such as *hamming loss*, *precision* and *recall*. We build upon this approach in three different ways. First we propose a single framework for both retrieval and ranking. This is accomplished by adopting a ranking approach similar to [74], where the output is a set of images ordered by relevance, enabling integration of ranking and reverse learning within the same framework. Secondly, we facilitate training, as well as retrieval and ranking, based on queries consisting of multiple-labels. In

[67], training and retrieval were performed independently for each label, whereas we explicitly utilize multi-labeled samples present in the training set for the purpose of learning our model. Finally, we model and learn the pairwise correlations between different labels(attributes) and exploit them for retrieval and ranking. We show that these improvements result in significant performance gains for both ranking and retrieval.

2.3.1 Retrieval

Given a multi-attribute query $\mathcal{Q}$, where $\mathcal{Q} \subset \mathcal{X}$, our goal is to retrieve images from the set $\mathcal{Y}$ that are relevant to $\mathcal{Q}$. Under the reverse learning formulation described above, for an input $\mathcal{Q}$, the output is the set of images y^* that contain all the constituent attributes in $\mathcal{Q}$. Therefore, the prediction function $f_w : \mathcal{Q} \rightarrow y$ returns the set y^* which maximizes the score over the weight vector w:

$$y^* = \arg\max_{y \subset \mathcal{Y}} w^T \psi(\mathcal{Q}, y) \tag{2.1}$$

here w is composed of two components; w^a for modeling the appearance of individual attributes and w^p for modeling the dependencies between them. We define $w^T \psi(\mathcal{Q}, y)$ as:

$$w^T \psi(\mathcal{Q}, y) = \sum_{x_i \in \mathcal{Q}} w_i^a \Phi_a(x_i, y) + \sum_{x_i \in \mathcal{Q}} \sum_{x_j \in \mathcal{X}} w_{ij}^p \Phi_p(x_j, y) \tag{2.2}$$

where

$$\Phi_a(x_i, y) = \textstyle\sum_{y_k \in y} \phi_a(x_i, y_k) \tag{2.3}$$

$$\Phi_p(x_j, y) = \textstyle\sum_{y_k \in y} \phi_p(x_j, y_k) \tag{2.4}$$

$\phi_a(x_i, y_k)$ is the feature vector representing image y_k for attribute x_i. $\phi_p(x_j, y_k)$ indicates the presence of attribute x_j in image y_k, which is not known during the test phase and hence $\phi_p(x_j, y_k)$ can be treated as a latent variable [79]. However, we adopt a simpler approach and set $\phi_p(x_j, y_k)$ to be the output of an independently trained attribute detector. In equation 2.2, w_i^a is a standard linear model for recognizing attribute x_i based on the feature representation $\phi_a(x_i, y_k)$ and w_{ij}^p is a potential function encoding the correlation between the pair of attributes x_i and x_j. By substituting (2.3) into the first part of (2.2), one can intuitively see that this represents the summation of the confidence scores of all the individual attributes x_i in the query $\mathcal{Q}$, over all the images $y_k \in y$. Similarly, the second(pairwise) term in (2.2) represents the correlations between the query attributes $x_i \in \mathcal{Q}$ and the entire set of attributes $\mathcal{X}$, over images in the set y. Hence, the pairwise term ensures that information from attributes that are not present in the query $\mathcal{Q}$, is also utilized for retrieving the relevant images.

Given a set of multi-label training images $\mathcal{Y}$ and their respective labels, our aim is to train a model w which given a multi-label query $\mathcal{Q} \subset \mathcal{X}$, can correctly predict the subset of images y_t^* in a test set $\mathcal{Y}_t$, which contain all the labels $x_i \in \mathcal{Q}$. Let $\mathbf{Q}$ be the set of queries; in general we can include all queries, containing a single attribute as well as multiple attributes, that occur in the training set. During the training phase, we want to learn w such that, for each query $\mathcal{Q}$, the desired output set of retrieved images y^*, has a higher score (equation 2.1) than any other set $y \in \mathcal{Y}$. This can be performed using a standard max-margin training formulation:

$$\arg\min_{w,\xi} \quad w^T w + C \sum_t \xi_t \tag{2.5}$$
$$\forall \;\; t \quad w^T \psi(\mathcal{Q}_t, y_t^*) - w^T \psi(\mathcal{Q}_t, y_t) \geq \Delta(y_t^*, y_t) - \xi_t$$

where C is a parameter controlling the trade-off between the training error and regularization, $\mathcal{Q}_t$ ($\mathcal{Q}_t \in \mathbf{Q}$) are the training queries, ξ_t is the slack variable corresponding to query $\mathcal{Q}_t$ and $\Delta(y_t^*, y_t)$ is the loss function. Unlike standard SVMs which use a simple 0/1 loss, we employ a complex loss function as it enables us to heavily(gently) penalize outputs y_t that deviate significantly(slightly) from the correct output y_t^*, measured based on the performance metric we want to optimize for. For example, we can define $\Delta(y_t^*, y_t)$ for optimizing training error based on different performance metrics as follows:

$$\Delta(y_t^*, y_t) = \begin{cases} 1 - \frac{|y_t \cap y_t^*|}{|y_t|} & \text{precision} \\ 1 - \frac{|y_t \cap y_t^*|}{|y_t^*|} & \text{recall} \\ 1 - \frac{|y_t \cap y_t^*| + |\bar{y}_t \cap \bar{y}_t^*|}{|\mathcal{Y}|} & \text{hamming loss} \end{cases} \tag{2.6}$$

Similarly, one can optimize for other performance measures such as F_β. This is the main advantage of the reverse learning approach, as it allows one to train a model optimizing for a variety of performance measures.

The quadratic optimization problem in Equation 2.5 contains $O(|\mathbf{Q}|2^{|\mathcal{Y}|})$ constraints, which is exponential in the number of training instances $|\mathcal{Y}|$. Hence, we adopt the constraint generation strategy proposed in [26], which consists of an iterative procedure that involves solving Equation 2.5, initially without any constraints,

and then at each iteration adding the most violated constraint of the current solution to the set of constraints. At each iteration of the constraint generation process, the most violated constraint is given by:

$$\xi_t \geq \max_{y_t \subset \mathcal{Y}} \left[\Delta(y_t^*, y_t) - (w^T \psi(\mathcal{Q}_t, y_t^*) - w^T \psi(\mathcal{Q}_t, y_t)) \right] \tag{2.7}$$

Equation 2.7 can be solved in $O(|\mathcal{Y}|^2)$ time, as shown in [67]. During prediction, we need to solve for 2.1, which again as shown in [67] can be efficiently performed in $O(|\mathcal{Y}| \log(|\mathcal{Y}|))$.

2.3.2 Ranking

We now show that, with minor modifications, the proposed framework for image retrieval can also be utilized for ranking multi-label queries. In the case of image ranking, given a multi-attribute query $\mathcal{Q}$, where $\mathcal{Q} \subset \mathcal{X}$, our goal is to rank the set of images $\mathcal{Y}$ according to their relevance to $\mathcal{Q}$. Unlike image retrieval, where given an input $\mathcal{Q}$, the output is a subset of the test images, in the case of ranking the output of the prediction function $f_w : \mathcal{Q} \rightarrow z$, is a permutation z^*, of the set of images $\mathcal{Y}$:

$$z^* = \arg \max_{z \in \pi(\mathcal{Y})} w^T \psi(\mathcal{Q}, z) \tag{2.8}$$

where $\pi(\mathcal{Y})$ is the set of all possible permutations of the set of images $\mathcal{Y}$. For the case of ranking, we make a slight modification to ψ by having:

$$w^T \psi(\mathcal{Q}, z) = \sum_{x_i \in \mathcal{Q}} w_i^a \hat{\Phi}_a(x_i, z) + \sum_{x_i \in \mathcal{Q}} \sum_{x_j \in \mathcal{X}} w_{ij}^p \hat{\Phi}_p(x_j, z) \tag{2.9}$$

where

$$\hat{\Phi}_a(x_i, z) = \textstyle\sum_{z_k \in z} A(r(z_k)) \phi_a(x_i, z_k) \tag{2.10}$$

$$\hat{\Phi}_p(x_j, z) = \textstyle\sum_{z_k \in z} A(r(z_k)) \phi_p(x_j, z_k) \tag{2.11}$$

with $A(r)$ being any non-increasing function and $r(z_k)$ being the rank of image z_k. Suppose we care only about the ranks of the top K images, we can define $A(r)$ as:

$$A(r) = \max(K + 1 - r, 0) \tag{2.12}$$

This ensures that the lower(top) ranked images are assigned higher weights and since $A(r) = 0$ for $r > K$, only the top K images of the ranking are considered.

During the training phase, we are given a set of training images $\mathcal{Y}$ and the set of queries, $\mathbf{Q}$, that occur among them. Unlike many ranking methods, which simply divide the set of training images into two sets - *relevant* and *irrelevant* - corresponding to each query and just learn a binary ranking, we utilize multiple levels of relevance. Given a query $\mathcal{Q}$, we divide the training images into $|\mathcal{Q}| + 1$ sets based on their relevance. The most relevant set consists of images that contain all the attributes in the query $\mathcal{Q}$, and are assigned a relevance $\text{rel}(j) = |\mathcal{Q}|$, the next set consists of images containing any $|\mathcal{Q}| - 1$ of the attributes which are assigned a relevance $\text{rel}(j) = |\mathcal{Q}| - 1$ and so on, with the last set consisting of images with none of the attributes present in the query and they are assigned relevance $\text{rel}(j) = 0$. This ensures that, in case there are no images containing all the query attributes,

images that contain the most number of attributes are ranked highest. While we have assigned equal weights to all the attributes, one can conceivably assign higher weights to attributes involving race or gender which are difficult to modify and lower weights to attributes that can be easily changed(e.g. *wearing sunglasses*). We use a max-margin framework, similar to the one used in retrieval but with a different loss function, for training our ranking model:

$$\begin{aligned} \arg\min_{w,\xi} \quad & w^T w + C \textstyle\sum_t \xi_t \\ \forall \;\; t \quad & w^T \psi(\mathcal{Q}_t, z_t^*) - w^T \psi(\mathcal{Q}_t, z_t) \geq \Delta(z_t^*, z_t) - \xi_t \end{aligned} \tag{2.13}$$

where $\Delta(z^*, z)$ is a function denoting the loss incurred in predicting the permutation z instead of the correct permutation z^*, which we define as $\Delta(z^*, z) = 1 - \text{NDCG@100}(z^*, z)$. The *normalized discount cumulative gain*(NDCG) score is a standard measure used for evaluating ranking algorithms. It is defined as:

$$NDCG@k = \frac{1}{Z} \sum_{j=1}^{k} \frac{2^{\text{rel}(j)} - 1}{\log(1+j)} \tag{2.14}$$

where rel(j) is the relevance of the j^{th} ranked image and Z is a normalization constant to ensure that the correct ranking results in an NDCG score of 1. Since NDCG@100 takes into account only the top 100 ranked images, we set $K = 100$ in Equation (2.12).

In the case of ranking, the max-margin problem (Equation 2.13) again contains an exponential number of constraints and we adopt the constraint generation procedure, where the most violated constraint is iteratively added to the optimization

problem. The most violated constraint is given by:

$$\xi_t \geq \max_{z_t \in \pi(\mathcal{Y})} \left[\Delta(z_t^*, z_t) - (w^T \psi(\mathcal{Q}_t, z_t^*) - w^T \psi(\mathcal{Q}_t, z_t)) \right] \tag{2.15}$$

which, after omitting terms independent of z_t and substituting Equations (2.9),(2.10),(2.14) can be rewritten as:

$$\arg \max_{z_t \in \pi(\mathcal{Y})} \sum_{k=1}^{100} A(z_k) W(z_k) - \sum_{k=1}^{100} \frac{2^{\text{rel}_{(z_k)}} - 1}{\log(1+k)} \tag{2.16}$$

where

$$W(z_k) = \sum_{x_i \in \mathcal{Q}_t} w_i^a \phi_a(x_i, z_k) + \sum_{x_j \in \mathcal{Q}_t} \sum_{x_j \in \mathcal{X}} w_{ij}^p \phi_p(x_j, z_k) \tag{2.17}$$

Equation (2.16) is a linear assignment problem in z_k and can be efficiently solved using the Kuhn-Munkres algorithm [76]. During prediction, Equation (2.8) needs to be solved, which can be rewritten as:

$$\arg \max_{z \in \pi(\mathcal{Y})} \sum_{k} A(r(z_k)) W(z_k) \tag{2.18}$$

Since $A(z_j)$ is a non-increasing function, ranking can be performed by simply sorting the samples according to the values of $W(z_k)$.

2.4 Experiments and Results

Implementation Details: Our implementation is based on the "Bundle Methods for Regularized Risk Minimization" BMRM solver of [87]. In order to speed up the training, we adopt the technique previously used in [79, 28], which involves replacing $\phi_a(x_i, y_k)$ in Equations (2.3),(2.10) by the output of the binary

attribute detector of attribute x_i for the image y_k. This technique is also beneficial during retrieval, as pre-computing the output scores for different attributes can be done offline, significantly speeding up retrieval and ranking.

2.4.1 Evaluation

Retrieval: We compare our image retrieval approach to two state-of-the-art methods: Reverse Multi-Label Learning (RMLL) [67] and TagProp [133]. Neither of these methods explicitly model the correlations between pairs of attributes and in the case of multi-label queries we simply sum up the per-attribute confidence scores of the constituent attributes. In case of TagProp, we use the σML variant which was shown to perform the best [133]. Furthermore, for multi-label queries, we found that adding up the probabilities of the individual words gave better results and hence we sum up the output scores, instead of multiplying them as done in [133]. In case of RMLL and MARR we optimize for the hamming loss by setting the loss function as defined in (2.6).

Ranking: In case of ranking, we compare our approach against several standard ranking algorithms including rankSVM [72], rankBoost [73], Direct Optimization of Ranking Measures(DORM) [74] and TagProp [133], using code that was publicly available[1]. Here again, for ranking multi-attribute queries, we add up the

[1]rankSVM `www.cs.cornell.edu/People/tj/svm_light/svm_rank.html`; rankBoost `http://www-connex.lip6.fr/~amini/SSRankBoost/`; DORM `http://users.cecs.anu.edu.au/~chteo/BMRM.html`; TagProp `http://lear.inrialpes.fr/people/guillaumin/code.php\#tagprop`

output scores obtained from the individual attribute rankers.

We perform experiments on three different datasets (1) Labeled Faces in the Wild(LFW) [71] (2) FaceTracer [78] and (3) PASCAL VOC 2008 [70]. We point out that there is an important difference between these datasets. While the LFW and FaceTracer datasets consist of multiple attributes within a single class i.e. *human faces*, the PASCAL dataset contains multiple attributes across multiple object classes. This enables us to evaluate the performance of our algorithm in two different settings.

2.4.2 Labeled Faces in the Wild (LFW)

We first perform experiments on the Labeled Faces in the Wild(LFW) dataset [71]. While, LFW has been primarily used for face verification, we use it for evaluation of ranking and retrieval based on multi-attribute queries. A subset consisting of 9992 images from LFW was annotated with a set of 27 attributes (Table 2.1). We randomly chose 50% of these images for training and the remaining were used for testing.

We extract a large variety of features for representing each image. Color based features include *color histograms, color corelograms, color wavelets* and *color moments*. Texture is encoded using *wavelet texture* and *LBP histograms*, while shape information is represented using *edge histograms, shape moments* and SIFT based visual words. To encode spatial information, we extract feature vectors of each feature type from individual grids of five different configurations (Fig. 2.2) and

Asian	Goatee	No Beard
Bald	Gray Hair	No Eyewear
Bangs	Hat	Senior
Beard	Indian	Sex
Black	Kid	Short Hair
Black Hair	Lipstick	Sunglasses
Blonde Hair	Long Hair	Visible Forehead
Brown Hair	Middle Aged	White
Eyeglasses	Mustache	Youth

Table 2.1: **List of Attributes**

concatenate them. This enables localization of individual attribute detectors, for example, the attribute detector for *hat* or *bald* will give higher weights to features extracted from the topmost grids in the configurations *horizontal parts* and *layout* (Fig. 2.2).

Figure 2.5 plots the NDCG scores, as a function of the ranking truncation level K, for different ranking methods. From the figure, it is clear that MARR (our approach) is significantly better than the other methods for all three types of queries, at all values of K. At a truncation level of 10 (NDCG@10), for single, double and triple attribute queries, MARR is respectively, 8.9%, 7.7% and 8.8% better than rankBoost [73], the second best method. The retrieval results are shown in Figure

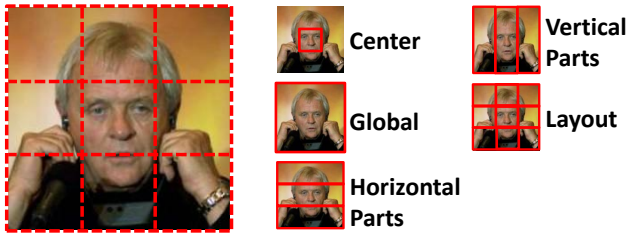


Figure 2.2: **Facial Feature Extraction:** Images are divided into a 3×3 grid(*left*) and features are extracted from five different configurations(*middle*,*center*).

2.3. In this case, we compare the mean areas under the ROC curves for single, double and triple attribute queries. Here MARR is 7.0%, 6.7% and 6.8% better than Reverse Multi-Label Learning (RMLL [67]), for single, double and triple attribute queries respectively. Compared to TagProp [133], MARR is 8.8%, 10.1% and 11.0% better for the three kinds of queries. Some qualitative results, for different kinds of queries are shown in Figure 2.4.

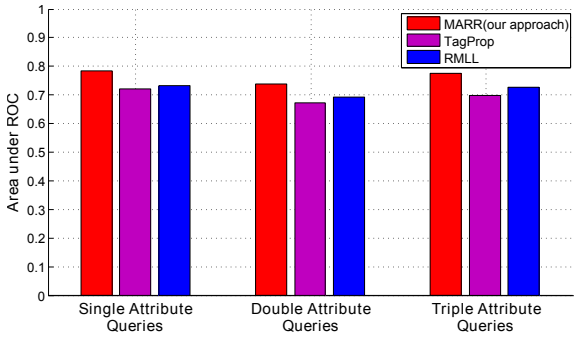


Figure 2.3: Retrieval Performance on the LFW dataset.

Figure 2.6 shows the weights learnt by the MARR ranking model on the LFW dataset. Each row of the matrix represents Equation 2.9 for a single-attribute query, with the diagonal elements representing w_i^a and the off-diagonal entries representing the pairwise weights w_{ij}^p. As expected, the highest weights are assigned to the

diagonal elements underlining the importance of the individual attribute detectors. Among the pairwise elements, the lowest weights are assigned to attribute pairs that are mutually exclusive such as (*White*,*Asian*), (*Eyeglasses*,*No-Eyewear*) and (*Short-Hair*,*Long-Hair*). Rarely co-occuring attribute pairs like (*Kid*,*Beard*), and (*Lipstick*,*Sex*) (*Sex* is 1 for male and 0 for female) are also assigned low weights. Pairs of attributes such as (*Middle-aged*,*Eyeglasses*) and (*Senior*,*Gray-Hair*) that commonly co-occur are given relatively higher weights. Also note that the weights are asymmetric, for example, a person who has a beard is very likely to also have a mustache, but not the other way round. Hence while retrieving images for the query "*mustache*", the presence of a beard is a good indicator of a relevant image, but not vice-versa, and this is reflected in the weights learnt.

2.4.3 FaceTracer Dataset

We next evaluate our approach on the FaceTracer Dataset [78]. We annotated about 3000 images from the dataset with the same set of facial attributes (Table 2.1) that was used on LFW. We represent each image by the same feature set and compare the performance of the ranking models learnt on the LFW training set. Figure 2.7 summarizes the results. One can observe that the performance of each method drops when compared to LFW. This is due to the difference in the distributions of the two datasets. For example, the FaceTracer dataset contains many more images of babies and small children compared to LFW. However, MARR still outperforms all the other methods and its NDCG@10 score is 5.0%, 8.1% and

11.6% better than the second best method(rankBoost) for single, double and triple attribute queries respectively, demonstrating the robustness of our approach.

2.4.4 PASCAL

Finally, we experiment on the PASCAL VOC 2008 [70] trainval dataset, which consists of 12695 images comprising 20 object categories. The training set consists of 6340 images, while the validation set consisting of 6355 images is used for testing. Each of these images have been labeled with a set of 64 attributes [69]. We use the set of features used in [69], with each image being represented by a feature vector comprised of edge information and color, HOG and texton based visual words.

Figure 2.9 plots the ranking results on the PASCAL dataset. We can observe that MARR substantially outperforms all other ranking methods except TagProp, for all the three kinds of queries. Compared to TagProp, MARR is significantly better for single attribute queries(7.4% improvement in NDCG@10) and marginally better for double attribute queries(2.4% improvement in NDCG@10), while TagProp is marginally better than MARR for triple attribute queries(1.5% improvement in NDCG@10). The retrieval results are shown in Figure 2.8, here, MARR outperforms TagProp by about 5% and Reverse Multi-Label Learning(RMLL [67]) by about 2%.

2.5 Conclusion

We have presented an approach for ranking and retrieval of images based on multi-attribute queries. We utilize a structured prediction framework to inte-

grate ranking and retrieval within the same formulation. Furthermore, our approach models the correlations between different attributes leading to improved ranking/retrieval performance. The effectiveness of our framework was demonstrated on three different datasets, where our method outperformed a number of state-of-the-art approaches for both ranking as well as retrieval. In future, we plan to explore image retrieval/ranking based on more complex queries such as scene descriptions, where a scene is described in terms of the objects present, along with their attributes and the relationships among them.

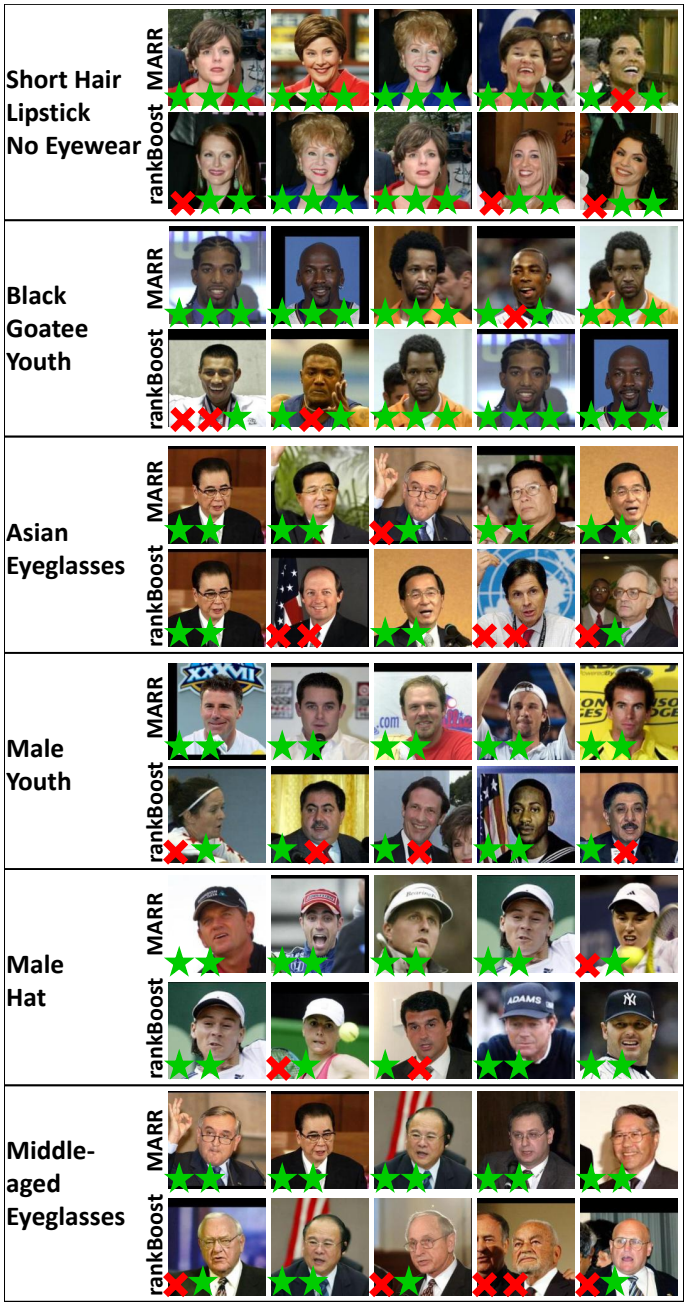
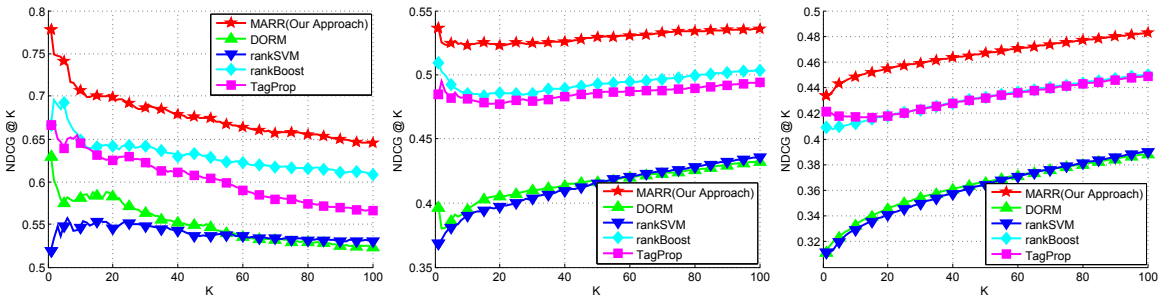


Figure 2.4: **Qualitative results:** Sample multi-label ranking results obtained by MARR and RankBoost(the second best method) for different queries on the LFW dataset. A *green star*(*red cross*) indicates that the image contains(does not contain) the corresponding attribute.



(a) Single Attribute Queries (b) Double Attribute Queries (c) Triple Attribute Queries

Figure 2.5: Ranking Performance on the LFW dataset

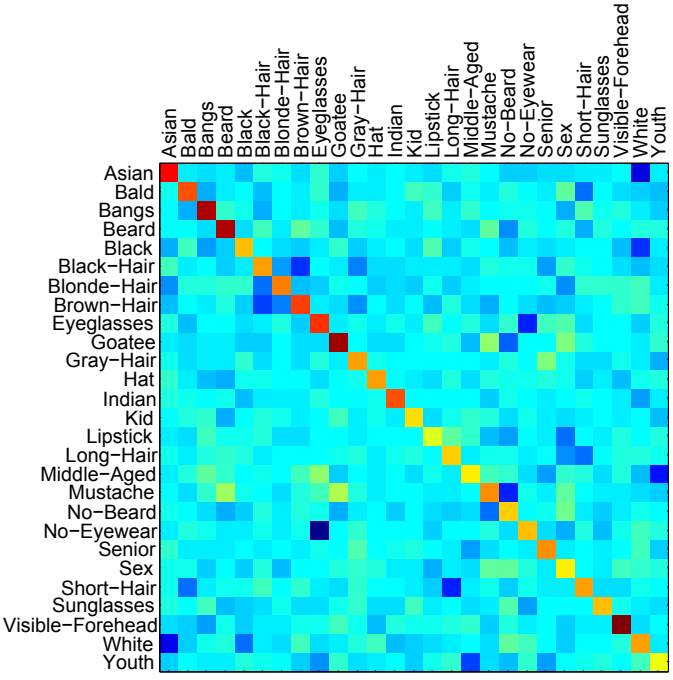
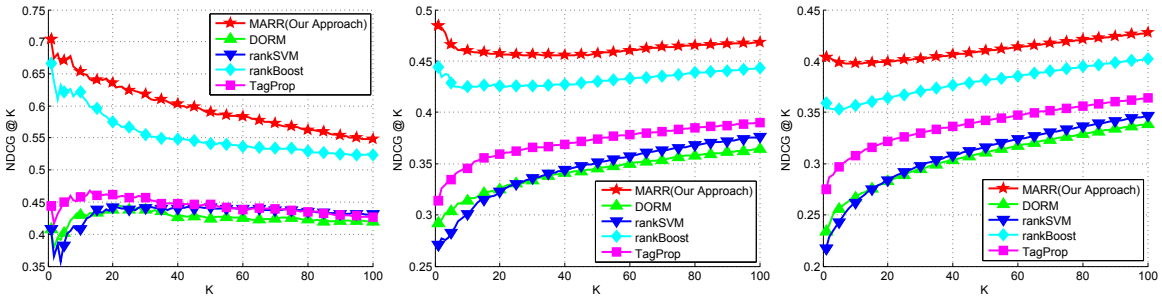


Figure 2.6: Classifier weights learnt on the LFW dataset, red and yellow indicate high values while blue and green denote low values. (best viewed in color).



(a) Single Attribute Queries (b) Double Attribute Queries (c) Triple Attribute Queries

Figure 2.7: Ranking Performance on the FaceTracer dataset

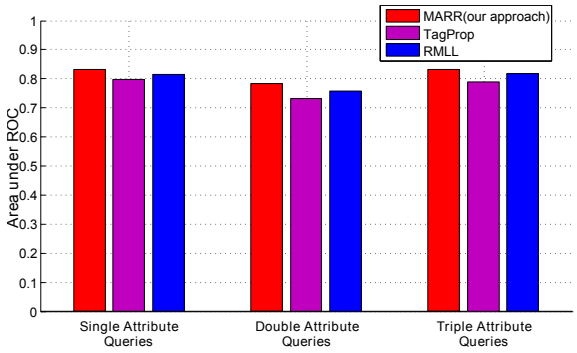
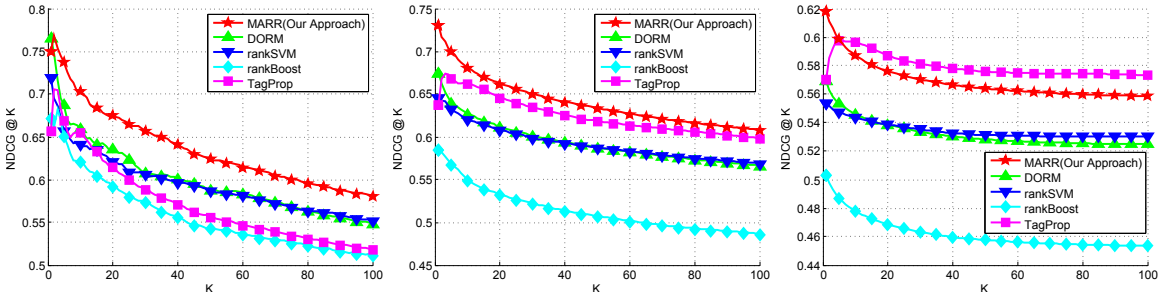


Figure 2.8: Retrieval Performance on the PASCAL dataset.



(a) Single Attribute Queries (b) Double Attribute Queries (c) Triple Attribute Queries

Figure 2.9: Ranking Performance on the PASCAL dataset.

Chapter 3

Multi-view Hashing for Multi-Modal Image Retrieval based on Complex Descriptive Queries

3.1 Introduction

The amount of visual data such as images and videos available over web has increased exponentially over the last few years and there is a need for developing techniques that are capable of efficiently organizing, searching and exploiting these massive collections. In order to effectively do so, a system, apart from being able to answer simple classification based questions such as whether a specific object is present(or absent) in an image, should also be capable of searching and organizing images based on more complex descriptive questions. To this end, there have been major advances in the field of information retrieval in the last few years. For example, search engines have become extremely adept at extracting and utilizing structured data from unstructured web pages and are even able to answer simple natural language based questions [145]. Similarly, image retrieval has progressed from retrieving images based on single label queries [81, 82] to multi-label queries [132, 133, 78, 117].

In this work, our goal is to enable a user to search for images based on significantly more descriptive queries that consist of *objects*, *attributes* - that further

describe properties of the objects; and *relationships* - that specify the relative configuration between pairs of objects. For example, we would like to search for images based on a query like (find an image containing a) *"red car to the left of a blue car which is in front of a blue bus"*. Clearly, such queries are significantly more expressive than multi-label queries and allow a user to search for more specific images/scenes based on certain characteristics of the scene.

While our framework affords a user significantly more expressive power than multi-label image retrieval approaches. It is also much more challenging to build, mainly due to two factors - Firstly the query is much more complex and the system should be able to correctly interpret the query. Secondly, a query can contain several constraints (e.g. object A should also contain attribute X or object A should be to the left of object B) and the system has to ensure that the retrieved images satisfy all of these constraints. We address these issues by adopting a spatial representation that is able to encode the locations and scales of different objects, their corresponding attributes and the relationships between them.

In image retrieval, queries are typically specified using an image, a sketch or a textual description and almost all current image retrieval approaches fall into one of these three categories. We attempt to combine these approaches by proposing a single framework thats allows the queries to be specified in either of these three modalities - i.e. images, sketches and text. In case of image based queries, the user provides an image as a query and would like to retrieve images that are semantically similar. The query image implicitly encodes the different objects present in the image, their attributes and the relationships between them. In the case of

a sketch based query, the user draws a sketch and explicitly labels the different regions with attributes and objects, while the locations and spatial relationships are automatically encoded within the sketch. Finally we have text based queries, where the objects, attributes and the relationships need to be explicitly provided by the user. However, building a large scale joint retrieval framework for multiple query modalities necessitates the ability to perform an efficient nearest neighbor search from a query of each modality to the elements in the database. We accomplish this, by proposing a multi-view hashing approach capable of hashing multiple views(modalities) of the query and the database elements to the same hash code. Our proposed multi-view hashing approach consists of a Partial Least Squares (PLS) based framework [146], to map queries from multiple modalities to a common linear subspace and are further converted into compact binary strings by learning a similarity preserving mapping, enabling scalable and efficient image retrieval from queries based on multiple modalities.

There are three main contributions of our work: **1)** We propose an approach for image retrieval based on complex descriptive queries that consist of objects, attributes and relationships. The ability to define a query by employing these constructs gives the user more expressive power and enables them to search for very specific images/scenes. **2)** Our approach supports query specification in three different modalities - images, sketches and text. Each of these modalities have their own advantages and disadvantages - for example, an image based query might provide the most information, but the user might not always have a query image; a text based query might not be specific enough; a sketch based query might require a

special interface. However, when equipped with the ability to search based on any of these modalities a user can choose the one that is the most appropriate for the given task. **3)** Finally, to support querying based on multiple query modalities, we propose a novel multi-view hashing approach that can map multiple views(modalities) of a query to the same hash code, enabling efficient image retrieval based on multi-modal queries.

3.2 Related Work

Image retrieval can be divided into three categories - image based retrieval, text based retrieval and sketch based retrieval - based on the modality of the query. In this work we propose an approach that integrates these approaches within a single joint framework. We now briefly describe relevant work in each of these image retrieval categories as well as relate and contrast our proposed approach to them.

In image based retrieval, the user provides a query in the form of an image and the goal is to retrieve similar images from a large database of images. A popular approach [121], involves utilizing a global image representation such as GIST or Bag-of-Words (BoW). Augmenting a BoW representation by incorporating spatial information has shown to improve retrieval results significantly [124, 144]. Further improvements have been obtained by aggregating local descripors [122] or by using Fisher kernels [123] as an alternative to BoW. However, a common drawback of these approaches is that, while they perform well at retrieving images that are visually

very similar to the query image (e.g. images of the same scene from a slightly different viewpoint), they can often retrieve images that are semantically very different. In contrast, we focus on retrieving images that are semantically similar to the query image.

Text based image retrieval entails retrieving images that are relevant to a text query, which in its simplest form could be a single word representing an object category. Early work in this area includes [81, 82]. Later work such as [132, 133, 78, 117], allowed for image retrieval based on multi-word queries, where a word could be an object or a concept as in [132, 133] or an attribute as in [78, 117]. Our work further builds upon these methods by providing a user the ability to retrieve images based on significantly more descriptive text based queries that consist of objects, attributes that provide additional descriptions of the objects and relationships between pairs of objects. While recent approaches such as [130, 131], do look at the problem of retrieving a relevant image given a sentence, they primarily focus on the reverse problem - i.e. producing a semantically and syntactically meaningful description of a given image.

Sketch based retrieval involves the user drawing a sketch of a scene and using it to search for images that have similar properties. An advantage of a sketch based query over text based queries is that it implicitly encodes the scale and relative spatial locations of the objects within an image. Initial approaches in sketch based retrieval include [125, 126], where the query was a color based sketch and the aim was to retrieve images that had a similar spatial distribution of colors. In [127], a sketch like representation of concepts called *concept map*, is used to search for

relevant images. In [129], Cao et al. have proposed an efficient approach for real-time image retrieval from a large database. However, their approach primarily relies on contours and hence uses information complementary to our method. In the field of graphics, people have looked at the problem of composing (rather than retrieving) an image from multiple images given a sketch [128].

Performing image retrieval in a large scale setting requires scalable approaches for compactly storing the database images in memory and efficiently being able to search for images relevant to the query in real-time. A popular approach consists of employing locality sensitive hashing (LSH) [134], which uses random projections to map the data into a binary code, while preserving the input-space distances in the Hamming space. Given a query, relevant images can be efficiently retrieved by computing the Hamming distance between the query and database images. Several recent approaches have also attempted to use the underlying data distribution to compute more compact codes [136, 137, 138, 135, 139, 140]. However, these approaches cannot be directly applied to our problem, since our queries can occur in multiple modalities (images, text and sketch). Therefore, we propose a novel multi-view hashing approach which builds upon [140] and allows multiple representations(views/modalities) to be mapped to the same binary code.

3.3 Approach

We now describe our approach for image retrieval that allows for multi-modal queries. This section is organized as follows: In subsection , we first define a sketch

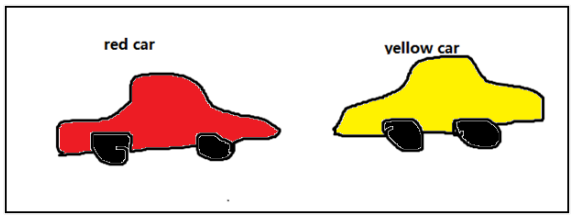


Figure 3.1: **sketch:** A sketch based query.

based query and a related semantic representation which facilitates compact encoding of the spatial relationships between the objects within an image. In subsections 3.3.2 and 3.3.3, we define image and text based queries and approaches for them into the semantic representation. Finally in subsection 3.3.4, we describe our proposed multi-view hashing approach which enables hashing different queries types to efficient binary codes.

3.3.1 Query Representation

We first define a sketch based query. As illustrated in Fig. 3.1, a sketch consists of a set of regions drawn by the user, with each region being labeled by an object class. A sketch can be thought of as a dense label map, with the unlabeled portions of the sketch belonging to the background class. Each region can also be labeled by multiple attributes, that could specify its color, texture or shape. We use sketches as our primary form of representation, and convert image and text based queries into sketches.

We convert the sketches into semantic representation that permits easy encoding of the spatial relationships between the objects in an image. The sketches are converted into C_o binary masks representing each object category and C_a masks

representing each attribute. The binary mask corresponding to the object o_k has value 1 at pixel (i, j) if the sketch contains the corresponding object class at pixel (i, j) and similarly for attributes. These binary masks are then resized to $d \times d$, leading to each sketch being represented by a vector of dimension $(C_o + C_a)d^2$. We compare the semantic similarity between two sketches based on the L_2 distance between their corresponding vector representations. This is similar to the semantic similarity metric used by [121], who utilize the spatial pyramid match [36] distance between corresponding label maps. Such a representation naturally encodes the scales and locations of different objects, the spatial relationships between pairs of objects within the image plane, and even the 3D relationships (through occlusion) to some extent. Our proposed semantic representation of objects and attributes bears some resemblance to the "Object Bank" [141] representation of Li et al.However, there is an important difference - while they use sparsity algorithms to tractably exploit the "Object Bank" representation, we instead rely on efficient hashing approaches to enable application of our representation to large scale problems.

3.3.2 image2sketch

In order to convert an image into a sketch, we semantically segment the image by assigning an object label to each pixel. The segmentation is performed using Semantic Texton Forests(STF) proposed by Shotton et al. [148]. We choose STFs over other semantic segmentation approaches primarily for their speed. Given a query image, STFs enable fast conversion of the image to the semantic feature

representation, which is critical for real-time image retrieval. Training the STF involves learning two levels of randomized decision forests - at the first level multiple randomized decision trees are learnt to cluster image patches into textons, with each leaf node representing a texton. The second level involves learning another tree that takes into account the layout and the spatial distribution of the textons to assign an object label to each pixel. During the test phase, the image patch surrounding each pixel is simply passed down each tree and the results of multiple trees are averaged to obtain its object label. We direct the reader to [148] for further details of the approach. We further improve the accuracy of the system by applying Image Level Priors (ILP), which is akin to utilizing the scene label obtained from a Spatial Pyramid Match [36] based scene classifier to influence and improve object detection performance. We also train STF based attribute classifiers and segment the image based on attributes. By semantically segmenting the image using STFs, we obtain the class and attribute label assignments for each pixel (sketch), which we then convert into the semantic representation, as described above.

3.3.3 text2sketch

We now describe an innovative approach for proposing a set of plausible candidate sketches relevant to a text based query. We assume that our query consists of a set of objects, with each object being described by zero or more attributes and a set of zero or more pairwise relationships between each pair of objects. An example of such a query is (find an image containing a) *"red car to the left of a blue car which*

is in front of a blue bus". We also assume that the text query has been parsed into its constituent components.

Corresponding to each object, we generate a large number of candidate bounding boxes by sampling from the training data. A bounding box X is defined by its scale and location (s_x, s_y, x, y). For each object o_i that is part of the query, we generate a set of bounding boxes $\mathcal{X}_{o_i}$ by importance sampling from the training data and also assign each bounding box a probability $P(X|c_i)$ (where c_i is the class of o_i) based on the training distribution. A candidate sketch of the query can be created by simply choosing one bounding box corresponding to each object o_i. However, to create semantically plausible sketches, we use the spatial relationship priors between pairs of object categories, learnt from the training data, as well as the knowledge of pairwise relationships provided by the user, in the query, to generate the set of most likely candidate sketches. We define the likelihood of a sketch as:

$$\begin{aligned} & P(X_{o_1}, X_{o_2}, ..|o_1, o_2, ..) \\ \propto & \prod_i P(X_{o_i}|c_i) \prod_{(j,k)} P(X_{o_j} - X_{o_k}|c_j, c_k) \end{aligned} \tag{3.1}$$

where X_{o_i} denotes the bounding box corresponding to object o_i, c_i is the object category of object o_i and $X_{o_j} - X_{o_k}$ represents the difference in the location and scale of the bounding boxes X_{o_j} and X_{o_k}. The first term in the equation represents the likelihood of an object of class c_i having a bounding box X_{o_i}, while the second term is a prior, that restricts the bounding boxes belonging to the pair of classes c_j and c_k from having arbitrary relative locations and scales. The second term is further

decomposed into its constituent components (s_x, s_y, x, y), as the priors on the joint distribution are not very accurate. This is similar to the contextual relationship model used by [2]. However, unlike [2], where the spatial relationships are binary, we employ a set of discrete bins to capture the degree of separation and relative scales between the objects within an image. We also incorporate information about the spatial relationships between a pair of object classes, contained within the query, into the model. For example, if the query states that object o_i is above object o_j, we renormalize $P(y_{o_i} - y_{o_j}|c_j, c_k)$ after setting $P(y_{o_i} - y_{o_j} > 0) = 0$. While we utilize relative relationships based only on scale and location, our model can be easily extended to incorporate relative orderings between attributes [2, 147], however we leave that for future work. Inference over this network is performed using loopy belief propagation.

We generate the set of N(=25) most likely candidate sketches based on the likelihood model (Eqn. 4.1). Chen and Weiss [142], have proposed an algorithm for finding the N-best configurations of a loopy model, which sequentially determines the next best configuration until the top N non-overlapping configurations have been identified. While this approach is directly applicable to our problem, we found that it did not work well in practice as it ends up finding a large number of configurations very similar to the best configuration. Hence we adopt the technique proposed by Park and Ramanan [143], which embeds a form of non-maximal suppression within the algorithm of [142] and results in a relatively diverse but at the same time highly likely set of candidate sketches. The candidate sketches for some sample queries are shown in Figs. 3.2, 3.3, 3.4.

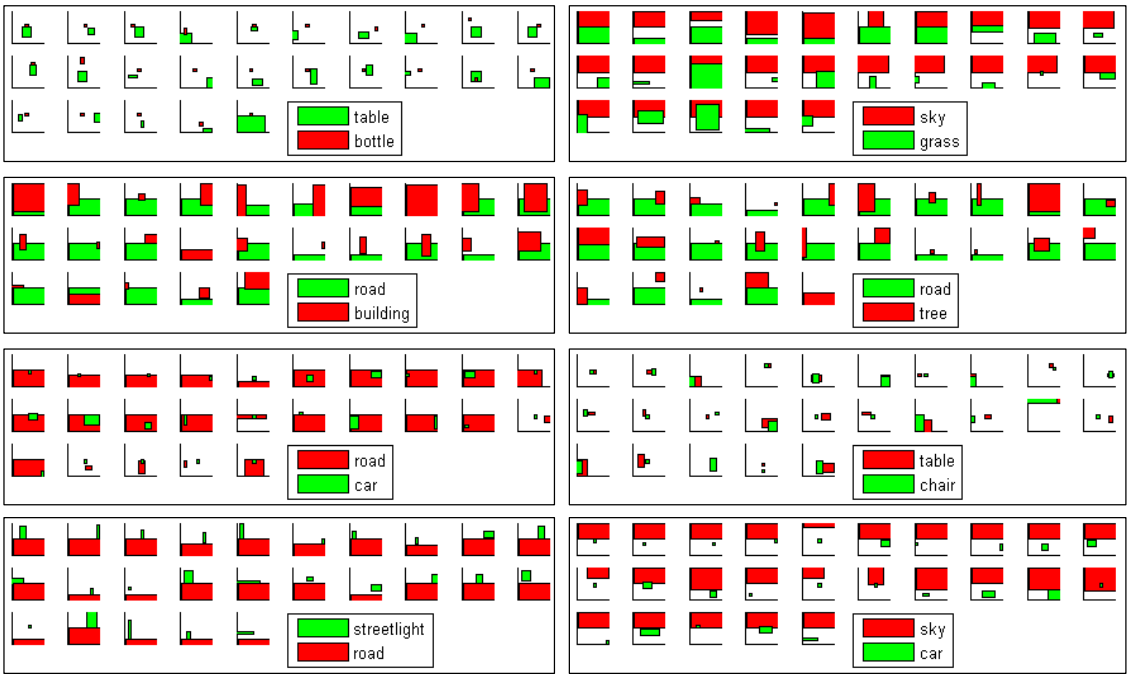


Figure 3.2: **text2sketch:** The top 25 sketches generated for text queries containing two objects with no relationship information.

3.3.4 Multi-Modal Hashing

We now describe our approach for multi-modal hashing. Here we are given a set of n data points, for which we have two different views $\mathcal{X} = \{\mathbf{x}_i\}, \ i = 1 \dots n, \ \mathbf{x}_i \in \mathcal{R}^{D_x}$ and $\mathcal{Y} = \{\mathbf{y}_i\}, \ i = 1 \dots n, \ \mathbf{y}_i \in \mathcal{R}^{D_y}$. For example, $\mathcal{X}$ could consist of the semantic representations computed from the images and $\mathcal{Y}$ could be the representations from the corresponding sketches. In general, we can have more than two views of the data. Our goal is to learn projection matrices W_x and W_y that can convert the data into a compact binary code. Where, the binary code $h_{\mathbf{x_i}}$ for the feature vector $\mathbf{x_i}$ is computed as $h_{\mathbf{x_i}} = \text{sgn}(\mathbf{x_i} W_x)$. Like most other hashing approaches, we want to learn W_x(and similarly W_y) that gives the same binary codes

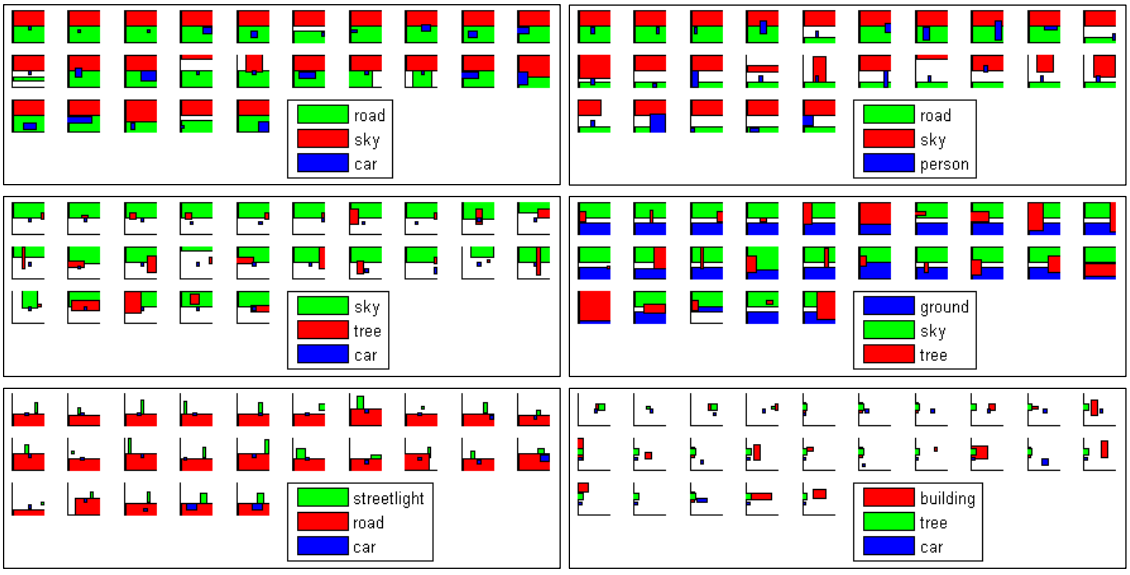


Figure 3.3: **text2sketch:** The top 25 sketches generated for text queries containing three objects with no relationship information.

$h_{\mathbf{x_i}}$ and $h_{\mathbf{x_j}}$ for data points $\mathbf{x_i}$ and $\mathbf{x_j}$ that are very similar. However, we also have the additional constraint that the projection matrices W_x and W_y produce similar binary codes $h_{\mathbf{x_i}}$ and $h_{\mathbf{y_j}}$ be the same when $\mathbf{x_i}$ and $\mathbf{y_j}$ are very similar. Motivated by the approach of [140], we adopt a two stage procedure - the first stage involves projecting different views of the data to a common low dimensional linear subspace, while the second stage consists of applying an orthogonal transformation to the linear subspace so as to minimize the quantization error when mapping this linear subspace to a binary code.

We adopt a Partial Least Squares (PLS) based approach to map different views of the data unto common latent linear subspace. We employ the PLS variant proposed in [146], which works by identifying linear projections such that the covariance between the two views of the data in the projected space is maximized.

Figure 3.4: **text2sketch:** The top 25 sketches generated for text queries containing two objects and a left/right relationship between the two objects. In the left column the red object is supposed to be to the left of the green object (according to the text query) and in the right column the red object is supposed to be to the right of the green object.

Let X be an $(n \times D_x)$ matrix containing one view of the training data $\mathcal{X}$, and Y be an $(n \times D_y)$ matrix containing the corresponding instances from a different view of the training data $\mathcal{Y}$. PLS decomposes X and Y such that:

$$\begin{aligned} X &= TP^T + E \\ Y &= UQ^T + F \\ U &= TD + H \end{aligned} \tag{3.2}$$

where T and U are $(n \times p)$ matrices containing the p extracted latent vectors, the $(D_x \times p)$ matrix P and the $(D_y \times p)$ matrix Q represent the loadings and the $(n \times D_x)$ matrix E, the $(n \times D_y)$ matrix F and the $(n \times p)$ matrix H are the residuals. D is a $p \times p$ matrix that relates the latent scores of X and Y. The PLS method iteratively constructs projection vectors $W_x = \{w_{x1}, w_{x2}, \ldots, w_{xp}\}$ and $W_y = \{w_{y1}, w_{y2}, \ldots, w_{yp}\}$ in a greedy manner. Each stage of the iterative process, involves computing:

$$[\text{cov}(t_i, u_i)]^2 = \max_{|w_{xi}|=1, |w_{yi}|=1} [\text{cov}(Xw_{xi}, Yw_{yi})]^2 \tag{3.3}$$

where t_i and u_i are the ith columns of the matrices T and U respectively, and $\text{cov}(t_i, u_i)$ is the sample covariance between latent vectors t_i and u_i. This process is repeated until the desired number of latent vectors p, have been determined. One can alternatively use CCA instead of PLS, however we found the performance of PLS to be slightly better than that of CCA, a conclusion that was also supported by [146].

PLS produces the projection matrices W_x and W_y that project different views of the data onto a common orthogonal basis. The first few principal directions computed by PLS contain most of the covariance, hence encoding all directions with the same number of bits results in a poor retrieval performance. In [140], the authors show that this problem can be overcome by computing a rotated projection matrix $\tilde{W}_x = W_x R$, where R is a randomly generated $(p \times p)$ orthogonal rotation matrix. Doing so, distributes the information content in each direction in a more balanced

manner, leading to the distances in the Hamming space better approximating the Euclidean distance in the joint subspace induced by PLS. They also propose a more principled and effective approach called Iterative Quantization (ITQ), which involves an alternating iterative optimization procedure to compute the optimal rotation matrix R, that minimizes the quantization error $\mathcal{Q}$, given by:

$$\mathcal{Q}(H, R) = ||H - XW_xR||_F^2 \tag{3.4}$$

where H is the $(n \times p)$ binary code matrix representing X and $||.||_F$ represents the Frobenius norm. Further details of the optimization procedure can be found in [140]. The effectiveness of the iterative quantization procedure for improving hashing efficiency by minimizing the quantization error has been demonstrated in [140]. Hence, we employ ITQ to modify the joint linear subspace for the multiple views produced by PLS and learn more efficient binary codes. The final projection matrices are given by $\tilde{W}_x = W_xR$ and $\tilde{W}_y = W_yR$, where R is obtained from (3.4).

3.4 Experiments and Results

We now present some preliminary results of our approach. We evaluate the performance of image and sketch based retrieval on the MSRC dataset, using Euclidean neighbors as ground truth. As in [140], we use the average distance to the 20th nearest neighbor to determine whether a database point returned for a given query is a true positive. Then, based on the Euclidean ground truth, we compute the precision-recall curves. The training and testing are performed on the MSRC

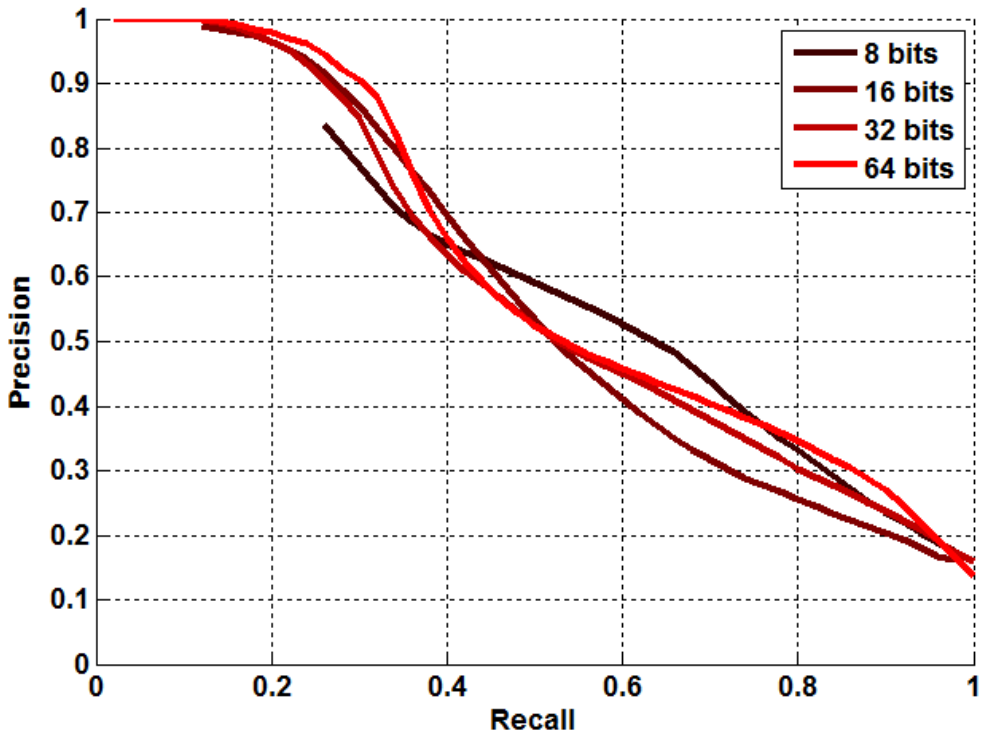


Figure 3.5: **sketch queries:**

training and test sets respectively. Fig. 3.5 shows the precision-recall curves obtained for different number of bits, for sketch based queries, while Fig. 3.6 plots the precision-recall curves using image based queries. It can be observed that in both the cases performance increases with the number of bits used for quantization, as expected, especially in high precision settings. In case of sketch based queries 3.5, the performance is higher as the error is only due to the quantization. Whereas in case of the image based queries, in addition to the quantization error, there is an additional error due to the prediction (*image2sketch*) which also adds up.

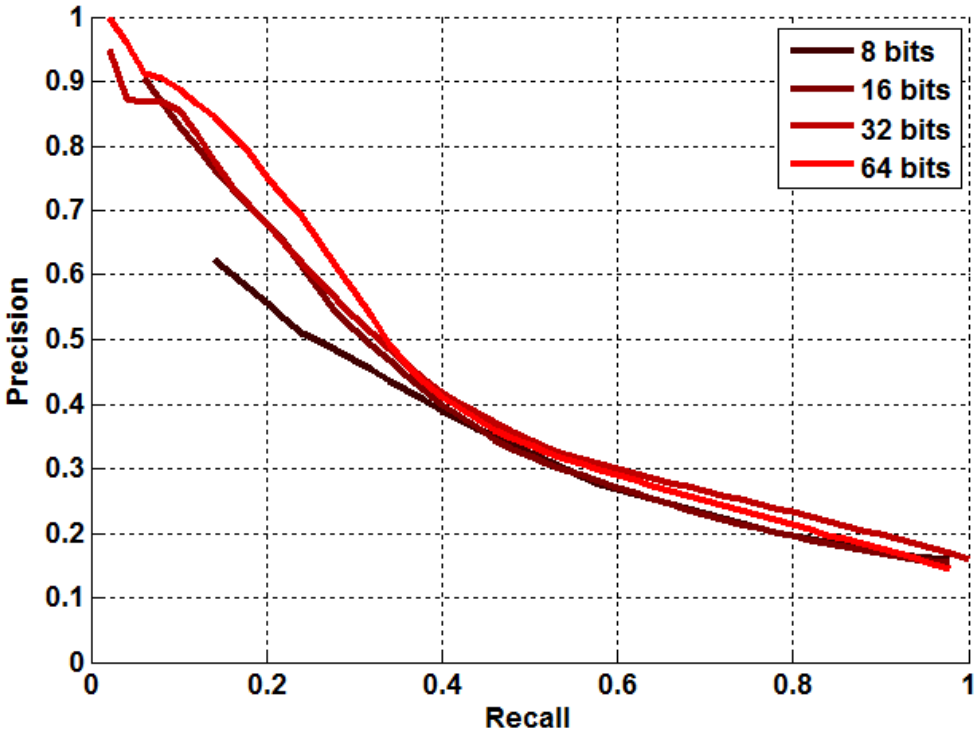


Figure 3.6: **image queries:**

3.5 Conclusion and Future Work

We have proposed an approach for multi-modal image retrieval based on complex descriptive queries that consist of objects, attributes and relationships. We have also proposed a unique multi-view hashing approach which maps semantically simlar queries of different modalities to the similar binary codes which enables application of our approach in large scale settings. We have evaluated our approach on a small dataset with promising results and we are currently working on a rigorous evaluation of this on a large scale database.

Chapter 4

Modeling Contextual Interactions for Multi-Class Active Learning

4.1 Introduction

Object recognition is one of the most challenging problems in computer vision. The performance of most recognition approaches, generally, depends upon the diversity and quantity of examples in the training dataset. There have been recent efforts aimed at gathering large training datasets [4, 1, 3]. However, these approaches have sought to obtain annotations for all the images in the dataset without prioritizing them on the basis of diversity. Such an approach leads to sub-optimal performance under finite/limited resources (manpower).

Due to the difficulty in obtaining a large amount of human labeling, many recent efforts have employed an active learning framework to choose regions to be labeled by human annotators. These approaches utilize the uncertainty in classification, asking humans to label examples which are hard to classify using the classifiers learned from previously labeled data. However, most of the work in active learning for visual recognition has focused on obtaining labeling for binary classification problems, especially where objects occur in isolation (such as the CALTECH-256 dataset). In the case of multi-class classification, these approaches seek to obtain the labels of high entropy regions.

We present a new framework for active selection of questions that simultane-

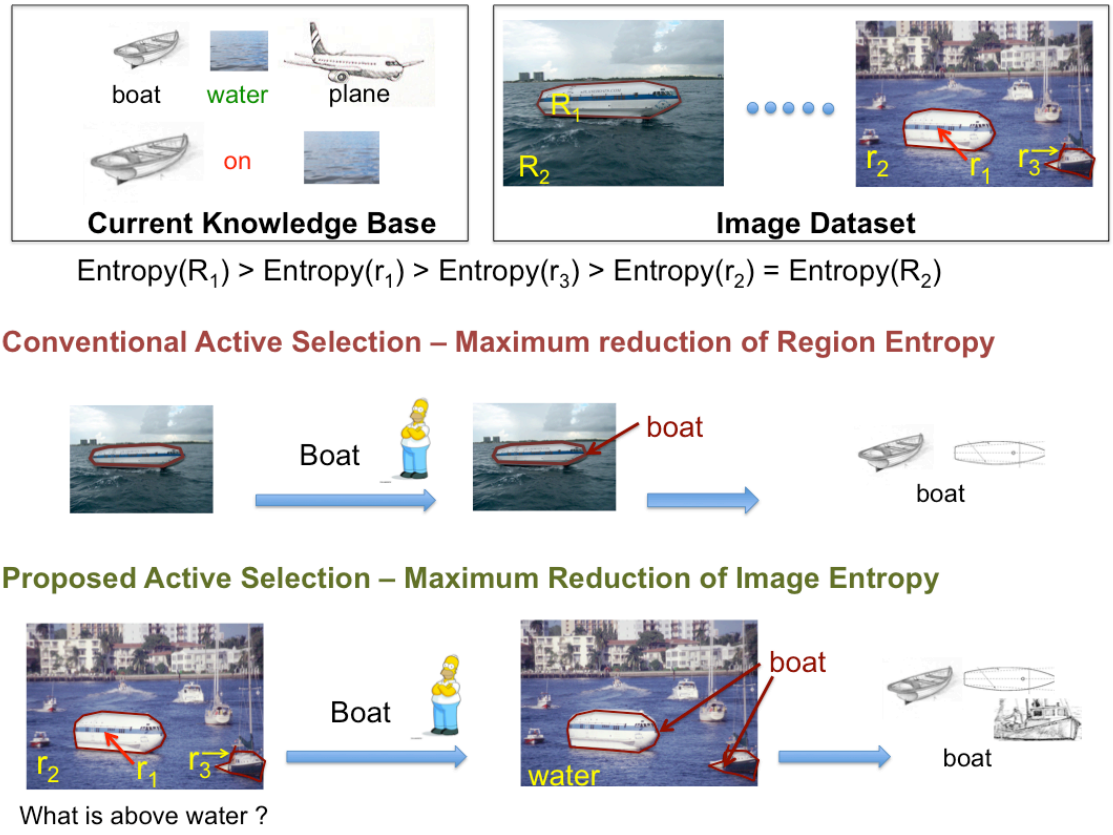


Figure 4.1: **Region Entropy vs. Image Entropy:** If we utilize region entropy only, region R_1 is selected for labeling since it has higher entropy than all other regions. Therefore, obtaining label of R_1 would lead to maximum reduction of entropy. On the other hand, if we consider image entropy and model the information yield due to contextual interactions, region r_1 is selected over R_1 since the label for r_1 would also provide information about other uncertain regions, such as r_3.

ously learns appearance and contextual models for scene understanding (multi-class classification) tasks. Our framework is based on active learning from natural images containing multiple objects. Traditionally, active learning approaches select questions which solicit the labels of uncertain regions. In contrast, we model contextual

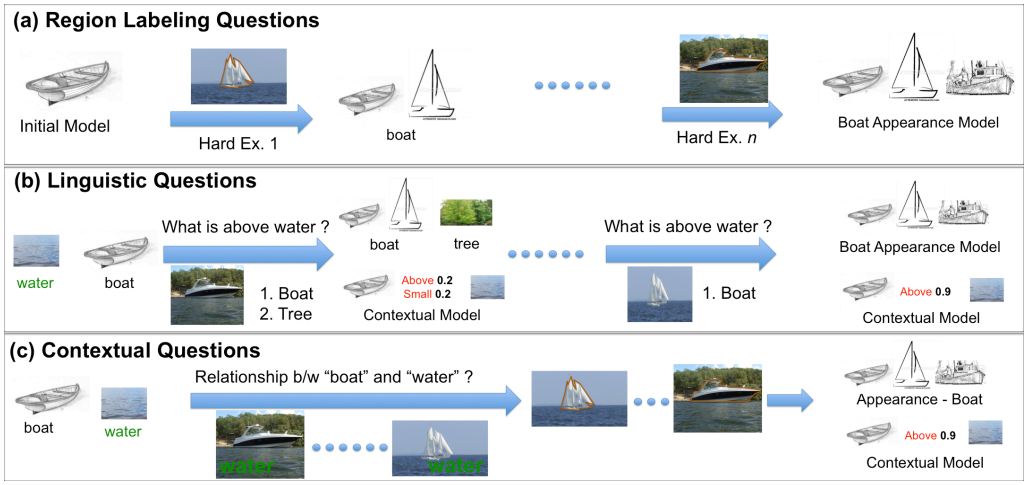


Figure 4.2: **Types of Questions:** Region labeling questions are the conventional questions utilized by active learning approaches. Here at each iteration the system asks the annotator to annotate the most uncertain region. Linguistic questions are questions which use the high confidence labels in the image to pose questions about uncertain regions. For example, since water is easy to recognize, the region associated with it is used to ask "what is above water". Contextual questions are the questions about contextual interactions between pair of objects in the world. For example, the system poses "what is relationship between boat and water". Contextual questions can be utilized to reduce the entropy of the all the training images since concepts can help in dis-ambiguating other uncertain regions.

interactions between image regions and solicit labels for those regions that yield significant reduction in the combined entropy of all the regions in the image (image entropy). Therefore, our criteria selects regions which are likely to yield information about the other confusing regions in the image as well. For example, consider the scenario shown in figure 4.1. Traditional active learning approaches would select

region R_1 to be labeled, since it is the most uncertain region. In contrast, our approach would evaluate the importance of each label not only based on the local region entropy, but also on how much new information that labeled region would provide about other uncertain regions in the image. Therefore, our approach selects r_1 since knowledge of r_1 label (boat) would yield information that would help reduce entropy of other regions, such as r_3.

One issue with using muti-object images for learning is localization of the objects of interest. Current active learning approaches handle this by either asking annotators to provide the boundaries or prompting labels on segmentations / super-pixels [21]. While such conventional labeling questions can be included in our active learning criteria, we also introduce linguistic questions which utilize additional constructs (such as prepositions or adjectives) in language for handling localization. In linguistic questions, the regions that can be linked to a concept with high confidence are used as anchors to ask questions about unknown regions in the scene. For example, in figure 4.2(b), the water region (easy to recognize) can be utilized as an anchor to ask questions such as " what is on the water?". Visual attributes of regions can also be used for anchoring, and lead to questions such as "What is the white region in the image?". These linguistic questions mimic the way humans solicit information to actively learn about their environment. These questions are also vital for obtaining labels when conventional labeling interfaces (mouse and screen) are not available [1].

[1] A typical example of this is an interaction between a robot and a human where robot asks questions to actively learn about the environment.

The contributions of our work are three-fold: (1) We introduce a new criteria for active selection of labeling questions based on reduction in the joint entropy of all the regions in the image (image entropy). By considering image entropy as opposed to the entropy of individual regions, we generate labeling questions which yield information about the region not only whose label is solicited, but other regions in the image as well. Experiments indicate that this criteria outperforms two baseline approaches by a wide margin. (2) We introduce linguistic questions in the active learning framework. In such questions, high confidence regions in the scene are used as anchors to pose questions about high entropy regions. (3) Finally, we introduce a new active learning framework which not only prompts for labels of regions but also poses questions about contextual concepts. For example, as shown in figure 4.2, our approach asks the annotator: " What is the relationship between boat and water? ". By learning contextual concepts directly from the annotator, we achieve reduction in global entropy over entire dataset. This leads to faster learning of appearance models, as the concept can be applied throughout the training dataset to obtain new training examples (See figure 4.2).

4.2 Related Work

There has been recent interest in utilizing humans as resources for gathering visual recognition datasets[4, 1, 3, 5, 6]. Some research has focused on generating human-friendly interfaces for labeling [1] or keeping human interest level high by formulating the labeling task as a game [6]. However, in most of these approaches the selection of regions/images to be labeled is mostly random. In machine learning,

active learning approaches [17, 18, 11, 10] are used to rank unlabeled points based on classification uncertainty- difficult examples are chosen for labeling. Criteria for selection include heuristics based on the version space of SVMs [10, 9], disagreement among classifiers [11] and expected informativeness [13, 12].

Early work on active learning in computer vision focused on obtaining binary labels of isolated objects. In multi-class scenarios, these approaches[14, 15, 16] extend the framework by utilizing multiple binary 1-vs-all classifiers. These approaches have two drawbacks: (1) They cannot compare the uncertainty in prediction of an example for two different binary subproblems, and hence cannot identify the classes that require more training data. (2) They assume localized object windows are available in the training dataset. These methods are appropriate for prioritizing labeling of isolated object datasets like CALTECH-256, but would fail for obtaining annotations where multiple objects occur in the same image.

More recent approaches attempted to overcome these two problems. Jain et. al [20] presented an approach for multi-class active annotation utilizing a probabilistic variant of K-Nearest Neighbors. However, they still utilize active learning for selection of images with isolated objects. Settles et. al [19] present an active learning formulation of multiple-instance learning, where localization of positive examples is not required. In a recent paper, Vijayanarasimhan et. al [21] present an active learning formulation where multiple type questions can be used - one type of question solicits location information by labeling of super-pixels. However, they consider only binary classification problems and not contextual interactions. Our work is also related to [22] which exploits the same-class and different-class relations

between multiple data-points for active learning. This framework [22], however, cannot be extended easily to include spatial interactions (such as above, below) and other relationships (such as bigger, brighter) between data-points.

4.3 Problem Formulation

4.3.1 Contextual Object Recognition Model

Our contextual object recognition model is based on the generative model used by Gupta and Davis [2]. In this approach, the authors represent contextual relationships between objects using constructs in language such as prepositions and comparative adjectives. Object appearance models are based on features of a region (mean RGB, x, y, convexity...) and relationship models are based on differential features (features extracted from pair of regions - for example, difference in brightness of two regions).

We briefly describe the generative model (see figure 4.3(a)) and refer the readers to the paper[2] for details: Each image is segmented into regions and each region is assumed to be associated to a noun node. Every pair of noun nodes is connected by a relationship edge. The relationship edge provides the constraints on the type of relationships that can exist between the nouns (based on priors learned from data – for example, sun should occur above water). Relationship edges also draw their likelihood from the differential features extracted from the pair of regions. For an image I, let I_j be the region appearance features for the jth region of the image, R_j, and I_{jk} be the differential features computed between regions R_j and R_k. Then,

the joint probability $P(n_1, n_2..|I)$ can be written as:

$$\begin{aligned} &= P(n_1, n_2..|I_1, I_2..I_{12}..C_A, C_R) \qquad (4.1) \\ &\propto \prod_i P(I_i|n_i, C_A) \prod_{(j,k)} \sum_{r_{jk}} P(I_{jk}|r_{jk}, C_R) P(r_{jk}|n_j, n_k) \end{aligned}$$

where n_j represents the noun associated with region R_j, r_{jk} is the relationship between regions R_j, R_k while C_A and C_R represent the parameters learned for noun and relationship models respectively.

The inference equation above consists of three terms: the first term is the noun likelihood term, which reflects how well the appearance of the regions matches the appearance of the noun-classes. The second term is a relationship likelihood term which indicates how well differential features match with relationship word models and the third term is the prior which restricts the possible relationships between pair of noun-classes. Inference over this network is conducted using belief propagation.

4.3.2 Active Learning

During active learning we pose one of the three types of questions to the user, and utilize the user's answer to update the existing object recognition model. Our objective at each stage, is to select the question, whose answer will lead to the maximum improvement in the current recognition model. The three types of questions are:

- **Regional Labeling Question:** This is the type of question used in tradi-

tional active learning methods for building visual classifiers. The user is simply asked to provide the label of a selected region in an image[Figure 4.2(a)].

- **Linguistic Question:** Motivated by the way humans actively learn about new objects using additional linguistic constructs, we propose a new type of active learning question. In this question, regions linked to "certain" concepts are used as anchors in the image to pose questions about other regions. For example, in the scenario shown in figure 4.2(b), a user is asked a question such as "what is above the water?", and is required to list the objects in the image which satisfy the question. The user simply answers "boat" and "tree" and does not specify which regions correspond to which objects in the answer.
- **Contextual Question:** The user is asked to provide the possible relationship between a pair of object classes, n_i and n_j. For each possible relationship the user also specifies whether the objects are positively or negatively related with respect to the relationship.

Compared to previous active learning methods [20, 22], which proceed by determining the best region to label next, our task is much more complex. We must identify both the type of question to ask and select the most (potentially) informative question from the set of possible questions of that type. The size of the set of possible questions, especially the linguistic questions, is much more larger than in traditional active learning methods.

Many active learning approaches use uncertainty/entropy as the criterion to choose the region to label. The region with the highest entropy is chosen based on

the assumption that fixing its label would lead to maximum reduction in the overall entropy of the system. These approaches, however, ignore the interactions between different regions in the image and the information a label provides about other regions in the image. In contrast, we consider contextual interactions and formulate the selection based on likely reduction of image entropy (entropy based on all the regions of the image). For computational reasons, we ignore the effect of fixing the label of a region in an image on the other unlabeled images. Some approaches [21] choose questions whose answers(labels) are expected to minimize the uncertainty over the entire unlabeled dataset. However, during each round of active learning, they require evaluating the uncertainty on the entire unlabeled dataset for each possible answer of every question. This is impractical in the case of large multi-class problems, more so in our case where the number of possible questions is much higher than in traditional active learning methods. In the following section, we describe the information-theoretic measure, based on Shannon entropy, to quantify information gain for a question.

4.3.2.1 Entropy of the system

Our training set consists of a set of images $\mathcal{I}$, of which a small subset $\mathcal{I}_L$ is completely labeled, while the remaining, much larger, subset $\mathcal{I}_U$, is unlabeled. We use $\mathcal{I}_L$ to learn the initial contextual object recognition model and then employ our active learning framework to ask the user conventional and linguistic questions about images from the unlabeled subset $\mathcal{I}_U$ along with contextual questions, while attempting to minimize the total entropy on $\mathcal{I}_U$(which we define below).

Equation 4.1, gives the probabilities of all possible class label assignments to the different regions of an image, while taking into account the contextual relations between them. We can use these probabilities to compute the joint entropy of an image:

$$H(I) = \sum_{(n_1, n_2..) \in N} -P(n_1, n_2..|I) \log(P(n_1, n_2..|I)) \tag{4.2}$$

Directly computing the joint entropy is impractical due to its computational complexity, hence we need to approximate it. An obvious approximation is the the first order entropy, which is the sum of the entropies of each region considered individually:

$$H_{fo}(I) = \sum_{I_j \in I} \sum_{n_j \in N} -P(n_j|I_j) \log(P(n_j|I_j)) \tag{4.3}$$

However, this completely ignores the contextual uncertainty of the system. Hence we use the second order approximation of the joint entropy, which is defined as:

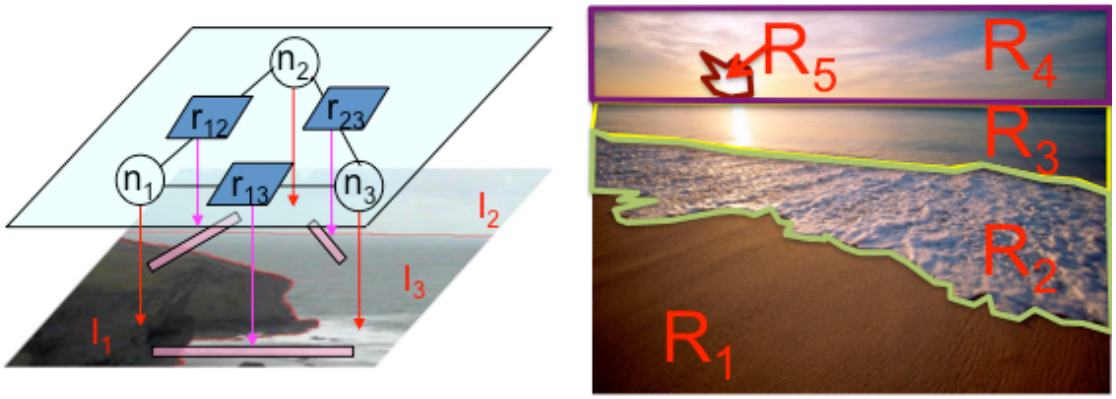
$$H_{so}(I) = \sum_{(I_j, I_k)} \sum_{(n_j, n_k) \in N} -P(n_j, n_k|I_j, I_k, I_{jk}) \log(P(n_j, n_k|I_j, I_k, I_{jk})) - (m-1) H_{fo}(I) \tag{4.4}$$

where m is the number of regions in the image I and $P(n_j, n_k|I_j, I_k, I_{jk})$ denotes the pairwise probability of regions R_j and R_k, which can be computed from Eqn. 4.1

assuming that the image contains only regions R_j and R_k. The total entropy of the system $H_{so}(\mathcal{I}_U)$, is then defined as the sum of the entropies of all the images, as they are independent of each other.

$$H_{so}(\mathcal{I}_U) = \sum_{I^i \in \mathcal{I}_U} H_{so}(I^i) \tag{4.5}$$

Based on this entropy measure, we define the importance of a question as the reduction in the system entropy resulting from knowing the answer to that question. Therefore, we compute the expected entropy reduction for each question and choose the one leading to the maximum expected reduction in entropy. We now describe the method for computing the expected entropy reduction for each type of question and the procedure for updating the current appearance and context models based on the answer obtained to each question.



(a) Graphical Model

(b) Linguistic Question: Example

Figure 4.3: (a) The graphical model used in [2]. (b) Linguistic Questions : An example of how certainty of some regions can be used to pose questions.

4.3.2.2 Region Labeling Questions

In region labeling questions, an annotator is prompted for the label of region R_j in image I. The expected reduction in the entropy of the image can be written as the reduction in entropy given that region R_j has the label c (and marginalizing over c). The reduction in entropy based on labeling the region R_j in image I is thus:

$$\Delta H_{so}(I, R_j) = \sum_{c \in \mathcal{C}} P(I_j|c, C_A)(H_{so}(I) - H_{so}(I|n_j = c))$$

where $H_{so}(I|n_j = c)$ denotes the entropy of the image, given that region R_j belongs to class c. After being labeled, the new class likelihood of region R_j is simply:

$$P(I_j|n_j) = \begin{cases} 1 & \text{if } n_j = c \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

Substituting the new likelihood $P(I_j|n_j)$, in (4.4), we obtain $H_{so}(I|n_j = c)$. Intuitively, it can be seen that in (4.4) $P(n_j, n_k|I_j, I_k, I_{jk}) = 0 \quad \forall n_j \neq c$ thereby decreasing the number of possible states of the image, leading to a reduction in its entropy. As the other images are independent of image I, $\Delta H_{so}(I, R_j)$ is also the total reduction in the system entropy. When the user provides the label(c) of region R_j, the corresponding features (I_j) are added to the training set and the appearance model of the class c is updated. Relationship priors are also updated based on the labels obtained.

4.3.2.3 Linguistic Questions

Linguistic questions utilize the high-confidence regions in the images and additional constructs (such as prepositions and comparative adjectives) in the language to ask labeling questions. For example, consider the image shown in figure 4.3(b). If one can recognize with certainty that region R_3 is water, then using this region as an anchor questions such as "what is above water ?" or "what is brighter than water?" can be posed.

We need to estimate the expected change in entropy for questions of the form: "What objects obey relationship r_k with respect to object A_c ?"(Expressed as $q = (r_k, A_c)$). The answer given by the user to this question is the list of classes $\mathcal{C}_q$ that satisfy the relationships. Let the regions that satisfy the relationship r_k w.r.t object class A_c in the image be represented by $\mathcal{R}_q$ (For example in fig.4.3(b), if $q = (above, water)$ then $\mathcal{R}_q = \{R_4, R_5\}$ since region R_3 is water). The entropy of the system is reduced since regions ($\mathcal{R}_q$) have a higher likelihood of belonging to the classes listed in $\mathcal{C}_q$. The new joint probability of the of the image is given by

$$P(n_1, n_2....|I, \mathcal{C}_q) = \sum_{\mathcal{R}_q} P(n_1, n_2....|I, n_{\mathcal{R}_q} \in \mathcal{C}_q) P(\mathcal{R}_q|I) \tag{4.7}$$

To compute $P(n_1, n_2....|I, n_{\mathcal{R}_q} \in \mathcal{C}_q)$, we modify the likelihood of the regions $\mathcal{R} \in R_q$ and recompute $P(n_1, n_2....|I)$ using equation 4.1. The new likelihoods are given by

$$P(I_j|n_j, C_A) = \begin{cases} 0 & \text{if } c \notin \mathcal{C}_q; \\ \frac{P(I_j|n_j, C_A)}{\sum_{c \in \mathcal{C}_q} P(I_j|n_j = c, C_A)} & \text{if } c \in \mathcal{C}_q \end{cases} \tag{4.8}$$

We also need to compute $P(\mathcal{R}_q|I)$. The set of regions that satisfy relationship r_k with anchor concept A_c in the image depends on the location of the anchor region R_{A_c} and the regions which satisfy relation r_k with the anchor region. Therefore, we can write it as:

$$P(\mathcal{R}_q|I) = \sum_{R_{A_c}} P(\mathcal{R}_q|r_k, R_{A_c}) P(I_{R_{A_c}}|A_c, C_A) \tag{4.9}$$

The new pairwise probabilities, $P(n_j, n_k|I_j, I_k, I_{jk}, \mathcal{C}_q)$ can be similarly computed. For a given answer $\mathcal{C}_q$, the entropy reduction is computed as:

$$\Delta H_{so}(q, \mathcal{C}_q) = H_{so}(I) - H_{so}(I|n_{\mathcal{R}_q} \in \mathcal{C}_q) \tag{4.10}$$

where $H_{so}(I|n_{\mathcal{R}_q} \in \mathcal{C}_q)$ denotes the new entropy of the image, which can be computed by substituting the new pairwise probabilities and the new likelihoods(Eqn. 4.8) in Eqn. 4.4.

The entropy reduction computed above depends on the answer, $\mathcal{C}_q$, to the question. However, at the time of selection the answer is not known. One could compute the entropy reduction for all possible sets of classes which could be the answer to the question and compute the expected entropy reduction as:

$$\Delta H_{so}(q) = \sum_{\mathcal{C}_q \in Pr(C)} P(\mathcal{C}_q|I) \Delta H_{so}(q, \mathcal{C}_q) \tag{4.11}$$

where $Pr(C)$ is the power set consisting of all possible combination of classes. This clearly is prohibitively expensive due to the large number of possible answers. Therefore, we employ importance sampling, where $\mathcal{C}_q$ is sampled based on the joint probability distribution computed from the current model.

The user answers a linguistic question by providing the list of class-labels $\mathcal{C}_q$ corresponding to the set of relevant regions. We can then compute the set of revised class probabilities for possible relevant regions from Eqn. 4.8, and then infer the classes of the regions using Eqn. 4.1, according to the new class probabilities. On obtaining the class assignments of the regions, we update the the appearance models of the corresponding classes by adding the regions to the training set. We also update the relationship priors $P(r_k|n_i, n_j)$ for the object pairs from the regions $\mathcal{R}_q$ and any other previously labeled regions in the image. Thus, linguistic questions, help in improving both the visual as well as the contextual components of our object recognition model.

4.3.2.4 Contextual Questions

In contextual questions, the annotator is asked for the relationships between a pair of object classes n_i and n_j, and he provides a list of possible relationships and whether these relationships occur "always" or "never". For example, if an annotator is asked : " List Relationship between sky and sea " then he can answer: "sky always occurs above sea and sky never occurs below sea".

For an object-object-relationship triplet the expected reduction in entropy can be obtained as:

$$\Delta H_{r_k,n_i,n_j} = \max \begin{cases} H_{so}(\mathcal{I}_U) - H_{so,high_{ijk}}(\mathcal{I}_U) \\ H_{so}(\mathcal{I}_U) - H_{so,low_{ijk}}(\mathcal{I}_U) \\ 0 \end{cases}$$

where $H_{so}(\mathcal{I}_U)$ denotes the entropy of the system according to the current model, given by Eqn. 4.5. $H_{so,high_{ijk}}(\mathcal{I}_U)$ denotes the system entropy under the assumption that the relation r_k positively holds between the object pair (n_i, n_j), which can be estimated by computing the system entropy with a modified contextual model where the relationship prior $P(r_k|n_i, n_j)$ is set to high. Similarly $H_{so,low_{ijk}}(\mathcal{I}_U)$ is the system entropy assuming that the relation r_k negatively holds between (n_i, n_j), and is obtained by computing the system entropy with $P(r_k|n_i, n_j)$ set to low. Here the assumption is that, if the current relationship priors do not accurately model a strong relationship(or the lack of it) between a pair of object classes, then correcting the relationship priors should result in a large reduction in the system entropy. Additionally, the entropy reduction will be relatively larger in the case of highly co-occurring object pairs, thereby favoring contextual questions on highly co-occurring pairs whose relationship priors are inaccurate. There can exist more than one strong relationship between an object pair, and representing each of them in the contextual model is important. Hence, we define the total expected entropy reduction of an object-pair as the sum of the entropy reductions due to all the individual relationships:

$$\Delta H_{n_i,n_j} = \sum_{r_k \in Rel} \Delta H_{r_k,n_i,n_j} \tag{4.12}$$

Computing the entropy reduction, for all pairs of object classes over the entire unlabeled dataset is, again, computationally expensive. To reduce the computational cost, we compute $\Delta H_{n_i,n_j}$ only from images in which the object pair (n_i, n_j) is expected to have a high joint likelihood. The joint likelihood in each image is determined from the current recognition model. The complexity can be further reduced by restricting the entropy reduction computation to only highly co-occurring object-class pairs, which can be determined from the expected co-occurrence over the entire dataset.

On obtaining the relationship labeling for the pair (n_i, n_j), the model is updated by setting the the positive relationship priors $P(r_{jk1}|n_i, n_j)..P(r_{jkc}|n_i, n_j)$, to a high value and the negative relationship priors to a low value.

4.4 Experimental Results

Implementation : Our appearance likelihoods are based on the approach in [20], which is a probabilistic variant of the K-nearest neighbor classifier, to model the likelihood of nouns. The relationship likelihood is modeled using a decision stump similar to [2]. The region and differential features used are the same as those used in [2]. Our relationship vocabulary consists of *above, below, left, right, more blue, more green, brighter*. For segmentation, we use the SWA algorithm [23] and perform stability analysis for estimating the stable segmentation level [24]. In all the experiments, the role of annotator is played by an Oracle which utilizes ground truth to obtain the answer to the questions.

We now present experimental results to demonstrate the effectiveness of our

active learning framework. We present a detailed experimental analysis of our approach on the MSRC dataset, along with additional results on the recently introduced Stanford dataset [7]. For evaluation, we compare our active learning framework to simple random sampling of questions and a state-of-the-art active learning method introduced in [20]. Both these baselines utilize only region labeling questions. We also provide individual performance of our constituent questions using the criteria based on image entropy as opposed to region entropy. Finally, we also present example questions of each kind posed by our system, showing that semantically meaningful and relevant linguistic as well as contextual questions are selected, further underlining their usefulness.

4.4.1 MSRC Dataset

We first show the performance of our approach on the standard MSRC dataset which consists of 532 images containing objects from 21 different categories. We use the standard training and test splits [8], consisting of 276 training images and 256 test images [2].

Ground Truth Segmentations: We first evaluate the performance of our approach under perfect segmentation by utilizing the ground-truth segmentations provided with the dataset. By isolating the errors due to segmentation, we can better understand the behavior of our active learning framework. A set of 34 fully-

[2]The generative model used in our work yields 72% recognition rate when trained using the perfect segmentations and the entire training set. This rate is comparable to state of the art approaches

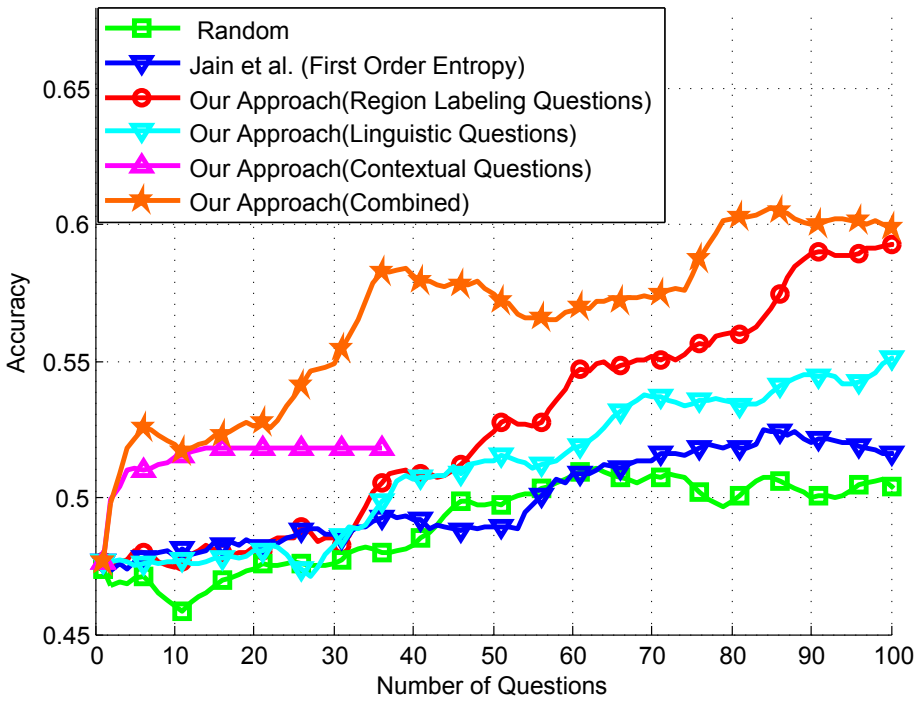


Figure 4.4: Performance on MSRC dataset when we utilize the ground truth segmentations of the images.

annotated images is chosen from the training set for building the initial model. Active learning is then used to improve the model by asking the user the three types of questions and using the response for updating the current model. Figure 4.4 shows the accuracy(region-level labels) of the different methods as a function of the number of questions answered, starting from the initial model.

It is clear from Figure 4.4 that our combined active learning framework is significantly better than the other methods. After 40 questions, our combined method has at-least a 14% improvement over all the other methods. As seen in the figure, utilizing a framework with different types of questions allows selection of the question-type which maximizes the entropy reduction. Therefore, initially our system asks contextual questions, since they reduce the entropy the fastest. This is

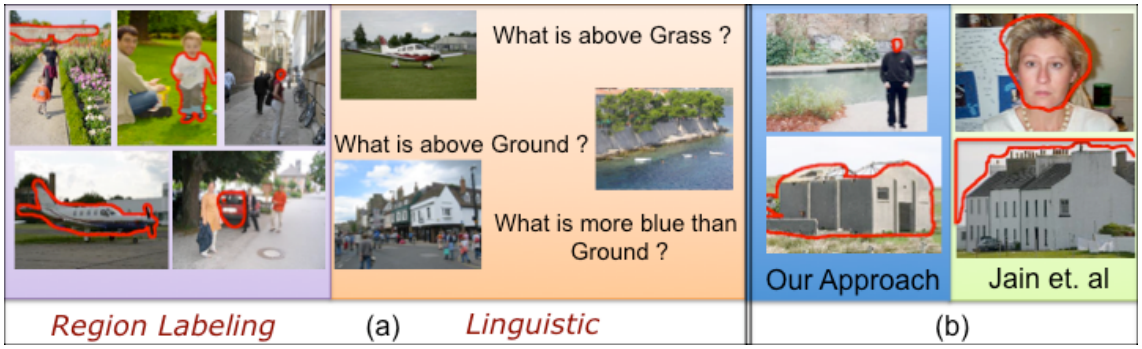


Figure 4.5: (a) A few examples of region labeling and linguistic questions posed by our framework in MSRC dataset with ground truth segmentations. Contextual questions posed by the system include: (1) What is relationship between grass and cow ? (2) What is relationship between sky and grass ? (3) What is relationship between tree and grass ? (b) Qualitative improvement in selection of questions.

generally followed by region labeling questions, which help in improving the appearance models. Once we have reasonably good appearance and context models, our system is able to find anchors to pose linguistic questions. The figure also shows the importance of utilizing image entropy over region entropy (Compare Region labeling curve to [20]). Utilizing the region labeling questions alone, our criteria outperforms both the random selection and the selection criteria proposed in [20]. Another interesting observation is that, as the number of unlabeled regions decreases the performance gain decreases (due to non-availability of informative questions).

Figure 4.5(a) shows some qualitative examples of question asked by our active learning framework. It can be seen how our system utilizes high confidence regions associated with grass, sky, ground to pose questions about other regions. Contextual questions asked by the system are also very important and relevant for

recognition. Figure 4.5(b) show some qualitative examples of improvement in selection by our framework. For example, [20] often selects regions from images where an object(face) occurs in isolation, based on the classification uncertainty of the region, for learning the appearance model. In contrast, our system selects regions(face) from images where other related regions(body) are also present, as fixing the label of those regions also provides information about the other labels. Another example of better selection is that while [20] selects regions such as sky to be labeled (in case of high uncertainty), our approach prefers to ask question or solicit labels about other regions in the image such as house. Fixing the house label also provides information about the region above. Since only tree or sky can occur above a house, the likelihood of confusing those regions with other objects decreases. Whereas fixing the sky label provides very less information about other regions in the image, as most objects generally occur below the sky.

Imperfect Segmentations: In this case we use a set of 50 fully-annotated images for the initial training and active learning is performed as described above. However, here the regions correspond to segments are automatically generated by the segmentation algorithm and this directly influences the region labeling and the linguistic questions that are selected. The evaluation of the test images is also performed based on the automatically generated segments. Figure 4.6 shows the accuracy of each method versus the number of questions answered. Here, again, it is clear that our method performs better than the other approaches. In case of imperfect segmentation the rate of increase of performance is slower. This is because ground-truth labels are provided only when the overlap between the segmentation

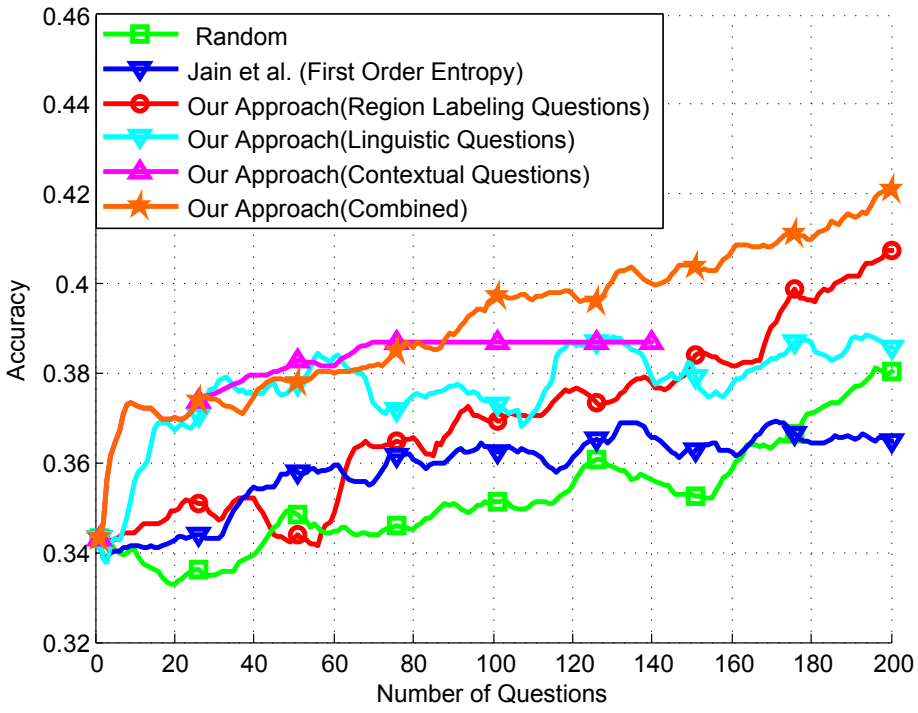


Figure 4.6: Performance on MSRC dataset using imperfect segmentations.

and ground truth region is high; otherwise the Oracle does not provide any answer to the question. In our experiments we found that approximately half of the regions were left unlabeled by the Oracle due to this reason. Furthermore, in case of imperfect segmentation the performance of linguistic questions saturates earlier. This is partly because of the poor performance of linguistic questions at the later stages when only the images with the absence of anchor regions remain (poorly segmented images).

4.4.2 Stanford Dataset

We also evaluate our approach on the Stanford dataset [7], which has been compiled from several already existing datasets and has accurate annotations collected using Amazon Mechanical Turk. It consists of 715 images, consisting of

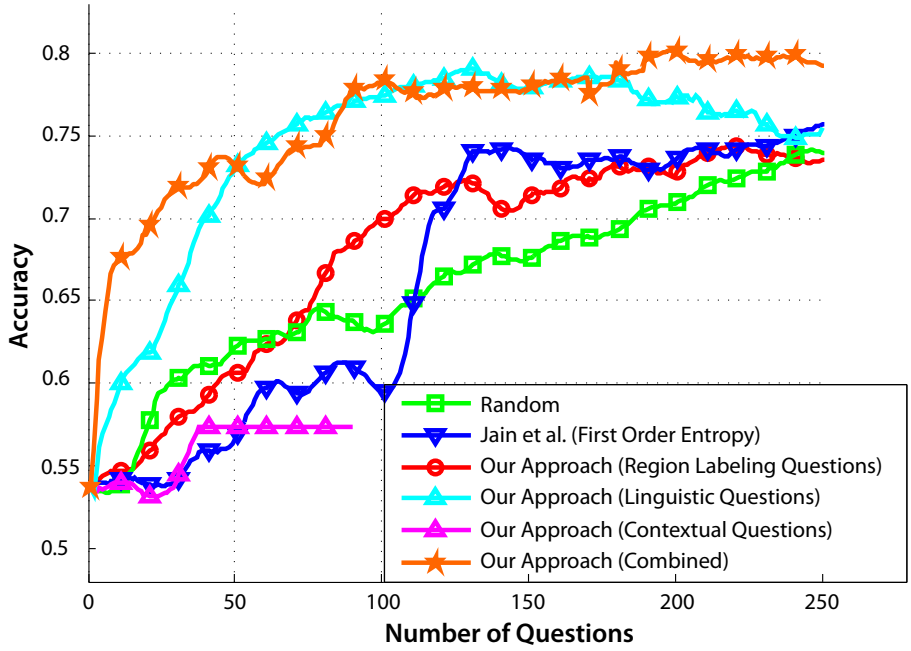


Figure 4.7: Performance of our system on Stanford dataset.

objects from 8 different categories. The images are randomly divided into a training set containing 415 images and a test set consisting of the remaining 300 images. A set of 8 images chosen from the training set, is used for building the initial model and active learning is employed for incrementally improving it. We consider only the top five regions(by area) in each image for both training as well as evaluation purposes. Figure 4.7 shows the accuracy(region-level labels) versus the number of questions, for each of the different methods. This dataset has 8 classes and therefore the initial context priors are very similar to final context priors and therefore contextual questions are not very helpful. However, due to good initial recognition rate our system finds anchors for linguistic questions more frequently and therefore linguistic questions outperform region labeling questions

Conclusion: We have presented an active learning framework that utilizes contextual interactions between regions in an image for selecting the regions to be labeled. Our criteria prefers regions which have high entropy and provide information about other regions in the image through contextual interactions. We present linguistic questions which utilize high confidence regions as anchors and additional constructs in language (prepositions, comparative adjectives) to pose questions about uncertain regions. Finally, our system can pose contextual questions and learn contextual concepts directly through an annotator.

Chapter 5

Unsupervised Transfer Learning for View-Invariant Object Detection

5.1 Introduction

Object detection and recognition is one of the core problems in computer vision. As a result, it has received considerable interest over the last few years. Many recognition approaches have been proposed and some of them [115, 104] have proven to be reasonably effective on relatively constrained datasets such as Caltech 101/256 and PASCAL VOC [111]. However, detection and recognition of objects in uncontrolled environments still remains an extremely challenging problem.

Here we focus on the problem of vehicle detection in urban surveillance environments. Traffic surveillance cameras are becoming increasingly widespread. Government agencies seek to use such cameras not just for monitoring traffic but also to search for suspicious vehicles, which requires accurate detection and localization of each vehicle. However, detection and localization of vehicles in surveillance video, which is typically low resolution, is extremely difficult as it requires dealing with view-invariance, varying illumination conditions (e.g. sunlight, shadows, reflections, rain, snow) and high density traffic situations, where vehicles tend to partially occlude each other.

There exist many methods for view-invariant object detection [104, 105, 98, 100, 101, 106, 110]. However, some of these approaches restrict themselves to learn-

ing appearance models for a small number of fixed viewpoints [104, 105] and often suffer a performance drop when presented with an unseen viewpoint. Although this issue can be overcome by learning models for a large number of viewpoints, doing so considerably slows down the detection speed as models for each viewpoint have to be evaluated. Likewise, methods that are capable of detecting objects from previously unseen viewpoints [98, 100, 101, 106, 110] are quite slow and therefore unsuitable for use in real-time applications.

In order to perform fast view-invariant object detection, we propose a novel approach which exploits scene layout and geometry to perform transfer learning in an unsupervised manner. Instead of building a view-invariant detector that can model all possible viewpoint deformations, which is extremely hard, we train simple object detectors for a large number of different viewpoints (source domains) that densely span the viewpoint space that we want to model. Given a new viewpoint(target domain), we exploit scene geometry and vehicular motion patterns to find closely related viewpoints from the source domain where vehicles are expected to occur in poses similar to the target viewpoint. Our dense representation in the viewpoint space ensures that we are guaranteed to find closely related viewpoints in the source domain. We then transfer the knowledge learnt a priori on the selected viewpoints for detecting vehicles in the new viewpoint. To match a new viewpoint to relevant viewpoints in the source domain, we learn a distance metric which, in addition to vehicle pose, also takes into account the generalizing ability of the detectors trained on the viewpoints in the source domain.

While our work is similar to methods such as [113, 114] which perform scene

annotation by directly transferring object category labels from similar scenes, we transfer richer information in the form of object detection models. Our approach also falls within the paradigm of using simple algorithms combined with large-scale training databases to achieve better results [4]. We demonstrate that a simple view-specific object detection method trained on a large but semantically organized dataset is able to outperform more sophisticated approaches.

The remainder of the chapter is organized as follows. We review related literature in Section 5.2, followed by a detailed description of our proposed approach in Section 5.3. We describe the experiments and results in Section 5.4 and finally conclude in Section 5.5.

5.2 Related Work

View-invariant object category recognition and detection has long been an important problem in computer vision [107]. Several methods address this problem by learning separate appearance models for a small number of canonical poses corresponding to each object category [104, 105, 98]. Other approaches such as [100, 101, 106, 110], employ richer parts-based-models, which learn the variation in appearance of the object parts as well the variation in the relationships between them over multiple views. Recently Gu and Ren [98] have proposed a discriminative approach for view-invariant object recognition based on a mixture-of-templates which also extends to the continuous case, and it has achieved the best performance on two different 3D object recognition datasets. However, a major disadvantage of

their approach is that, depending on the number of templates used, it can be up to an order of magnitude slower than a comparable view-specific object recognition method that employs a similar feature representation.

The presence(or absence), location and scale of objects in a scene are heavily influenced by the surrounding objects and the geometric layout of the scene. Several recent works take advantage of this fact to improve object detection results [97, 108, 99]. Hoiem et al. [97], proposed a joint framework for object detection and scene geometry estimation, where the scene layout helps refine object hypotheses and vice-versa. Similarly, Bao et al. [99] jointly infer the 3D object locations and 3D orientations of planar surfaces in the scene by utilizing the fact that the pose of an object is constrained by the orientation of the 3D plane upon which it rests. Our work follows a similar line; we use 3D scene geometry to infer expected object pose, which is effectively exploited to improve the speed and accuracy of view-invariant object detection.

Most supervised learning approaches assume that the training and the test data are drawn from the same distribution. However there are often cases when this assumption is violated, for example when the training and test data belong to different domains, resulting in a sharp drop in performance. The goal of transfer learning [96] is to address such issues by developing effective mechanisms for the transfer of knowledge between different but related domains. However, a large majority of work on transfer learning has focused on a supervised setting [92, 93, 94, 95], where the underlying assumption is that there is access to a large amount of out-of-domain(source domain) labeled training data and also a small amount of labeled

in-domain(target domain) training data. These methods can be divided into two categories - the first type of methods learn a complete model on the source domain and adapt it to the target domain by utilizing the available annotated target domain data [93, 94]. The second attempt to learn a cross-domain mapping between the source and target domains [92, 95]. In contrast, our application setting is completely unsupervised, as we do not have access to any annotations or even unlabeled data in the target domain[1]. A related problem has been previously addressed by Blitzer et al. [109] in the NLP domain, where they utilize structural correspondences between a pair of domains to first align and then transfer a model from the source domain to the target domain. Our proposed framework is based on a conceptually similar principle, as we utilize scene geometry and layout to identify an appropriate source domain for transferring an object recognition model to a specific target domain.

5.3 Unsupervised Transfer Learning

5.3.1 Training Dataset Collection

We have collected more than 400 hours of video from 50 different traffic surveillance cameras, located in a large North American city, over a period of several months. We adopt a simple method to extract images of cars from these videos, for training our object detection models. We perform background subtraction and

[1]During the training phase, we utilize labeled data from the source domain; No data(not even unlabeled) from the target domain is used. We refer to this setting as Unsupervised Transfer Learning.

obtain the bounding boxes of foreground blobs in each video frame. We also compute the motion direction of each foreground blob using optical flow. Vehicles are then extracted using a simple rule-based classifier which takes into account the size and motion-direction of the foreground blobs. The range of acceptable values of the size and motion-direction are manually specified for each camera view. We manually remove the accumulated false positives. This simple procedure enables us to collect a large number of images of vehicles(about 220 000) in a variety of poses and illumination conditions, while requiring minimal supervision. We utilize the motion direction of each foreground blob for categorizing the images of vehicles of each camera viewpoint into a set of clusters. Subsection 5.3.2 describes the process of obtaining the clusters. The clustering of images leads to categorization of the training images into a two level hierarchy, where the first level of categorization is according to the camera viewpoint and the second level is based on the motion-direction within each camera viewpoint. Since all the camera viewpoints are distinct, each leaf node of our hierarchy consists of training images of vehicles in a distinct pose. On an average, each camera viewpoint has about two clusters, resulting in a total of about 100 clusters(leaf nodes of the hierarchy) which is an extremely diverse collection of vehicles in different poses.

5.3.2 Object Pose Parametrization

We parametrize the pose of the vehicles within each cluster, in terms of their zenith (ϕ) and the azimuthal angles (θ) with respect to the camera. The zenith angle

can be estimated based on the position of the horizon and the azimuthal angle can be approximated by the motion direction of vehicles with respect to the camera.

Horizon Estimation: We estimate the position of the horizon in each camera view. Since our task is that of detecting vehicles in a traffic surveillance setting, our images consist of urban environments, which enables us to utilize the inherent structure present in such scenes to infer their 3D geometry. Several approaches [91, 90] have been proposed for estimating the position of the horizon in an image by exploiting the fact that urban scenes contain multiple sets of parallel lines which intersect at different vanishing points and that the horizon should pass through these vanishing points. We use the recently proposed, geometric image parsing approach [90][1] by Barinova et al. which has attained the best performance on the task of horizon estimation, on two different urban datasets. Figure 5.1 demonstrates the horizon estimation on two different images.

Motion Pattern Estimation: For each camera viewpoint, we estimate the direction of motion of vehicles appearing in that scene. For this purpose, we collect a five minute($\sim$ 9000 frame) video clip of the scene. We found that a clip of this duration is sufficient for capturing the regular motion patterns that occur at an intersection. We follow an approach similar to that of Yang et al. [102], who employ a clustering based method for discovering motion patterns in video. We first compute the optical flow of each frame in the video and represent each space-time point by a four dimensional vector consisting of the location of the point in the image plane and

[1]code: `http://graphics.cs.msu.ru/ru/science/research/3dreconstruction/geometricparsing`

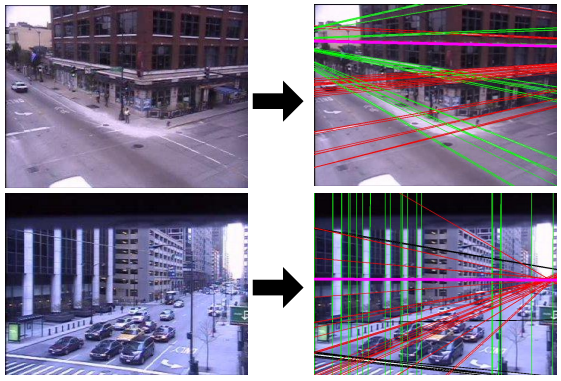


Figure 5.1: **Horizon Estimation:** Horizon estimation in urban scenes using [90]. The red and green lines represent groups of parallel lines, while the thick pink line represents the horizon.(best viewed in color).

the magnitude and direction of its optical flow - (x, y, v, θ). Points with an optical flow magnitude above or below certain fixed thresholds are assumed to be noise and are discarded. We randomly subsample the remaining points and cluster them using a self-tuning variant of spectral clustering [103], which automatically selects the scale of analysis as well as the number of clusters. The clusters so obtained represent the different directions of motion of vehicles appearing in the scene. We represent each cluster by the dominant direction of motion of the points within it and by its location in the image plane. The entire process is illustrated in Figure (5.2).

The pose of a vehicle can be defined in terms of its azimuthal angle θ and the zenith angle ϕ with respect to the camera. We assume there is no camera roll, as it can be easily rectified based on our estimation of the horizon. One can represent the variation in the pose of vehicles within a particular motion cluster of a camera

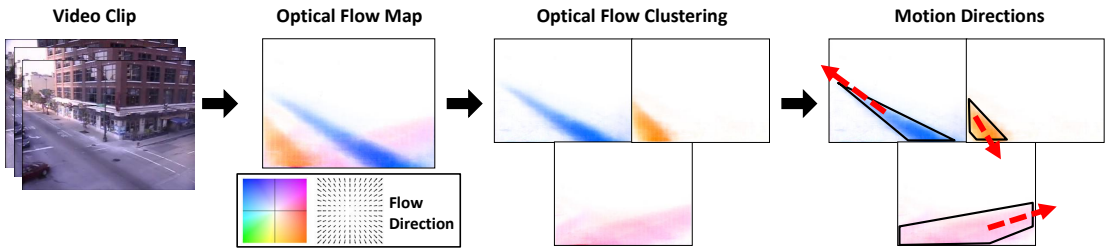


Figure 5.2: **Motion Pattern Estimation:** Given a camera view, we estimate its motion patterns from a short video clip by first computing its optical flow and clustering points in space-time based on their location and optical flow direction and magnitude. The resulting clusters represent the patterns of movement of vehicles; each motion pattern is represented by its dominant motion direction and location in the image plane.(best viewed in color).

viewpoint, in terms of the ranges of the zenith and azimuthal angles of the vehicles appearing in it. We define (u_c, v_c) as the optical center of the camera in the image plane and v_0 as the y-coordinate of the horizon. Let v_{min} and v_{max} respectively denote the upper and lower extent of a cluster in the $y-$direction (Figure 5.4), then the range of zenith angles ϕ (Figure 5.3) of vehicles appearing in that cluster can be defined as:

$$\phi_{\max} = \tan^{-1}\left(\frac{v_{\max}-v_c}{f}\right) + \tan^{-1}\left(\frac{v_c-v_0}{f}\right) \tag{5.1}$$

$$\phi_{\min} = \tan^{-1}\left(\frac{v_{\min}-v_c}{f}\right) + \tan^{-1}\left(\frac{v_c-v_0}{f}\right) \tag{5.2}$$

where f is the focal length of the camera. Here the assumption is that the optical center of the camera (v_c) lies below the location of the horizon in the image plane (v_0). The equations are similar in case the reverse is true. Note that these

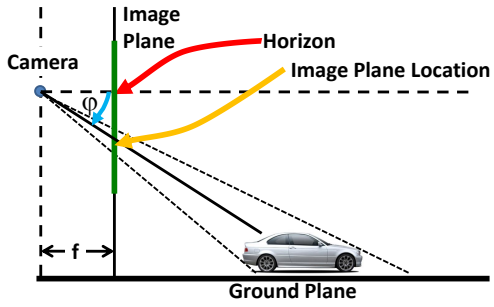


Figure 5.3: **Zenith Angle:** The zenith angle of a vehicle with respect to the camera.

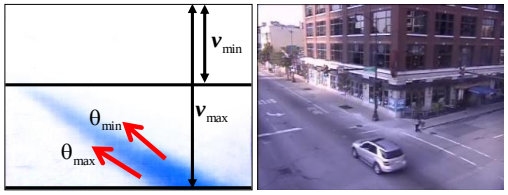


Figure 5.4: **Camera Viewpoint Parametrization:** The range of the azimuthal angles of a vehicle with respect to the camera $(\theta_{max}, \theta_{min})$. v_{max} and v_{min} denote the maximum and minimum y-coordinates of the motion cluster respectively and determine the range of the zenith angles of vehicles in the motion cluster (Equation 5.1,5.2).

equations are valid even when the image plane is not perpendicular to the horizon. We also compute the maximum (θ_{max}) and minimum (θ_{min}) directions of motion of vehicles with respect to the camera, based on the optical flow, and use them to approximate the azimuthal angles of vehicles within the motion cluster (Figure 5.4). Hence the pose of the vehicles of appearing in a cluster c_i can be represented in terms of the range of their zenith angles with respect to the camera $(\mathbf{A_i} = [\phi_{max}\ \phi_{min}])$ and the range of the direction of motion with respect to the camera $(\mathbf{Z_i} = [\theta_{max}\ \theta_{min}])$.

5.3.3 Transferring Object Detection Models

During the training phase we build models for recognizing vehicles in a variety of poses that are present in different camera viewpoints (source domains). As described in subsection 5.3.1, our training dataset has been categorized into a two level hierarchy, with each leaf node representing vehicles traveling in a specific direction as seen from a particular camera viewpoint. We train a Deformable Parts Model (DPM) [104] based object detector DPM_s corresponding to each leaf-node cluster c_s. While we chose DPM based detectors because they have consistently achieved the best performance on several object recognition benchmarks [111], our approach allows for using any off-the-shelf object recognition system, including the Viola-Jones object detector [50] which would enable usage in real-time applications.

Given a video captured from a previously unseen camera viewpoint (target domain), we first estimate the position of the horizon and compute the motion patterns of vehicles appearing in the scene. Corresponding to each cluster c_i, we then compute the range of azimuthal angles $\mathbf{A_i}$ and the range of zenith angles $\mathbf{Z_i}$. Since our source data contains a large number of camera viewpoints each of which contains vehicles moving in multiple directions, we have DPM based object detectors trained for a large number of possible poses. Furthermore, most object detectors are capable of handling a small degree of view invariance. Hence for each motion cluster c_i in the target view, we simply select the object recognition model from the source view that is likely to contain vehicles in the same pose and directly use it to detect vehicles in the target view. As discussed earlier, the vehicle pose is a function of the

direction of motion of vehicle with respect to the camera $\mathbf{A_i}$ and the zenith view direction $\mathbf{Z_i}$. While choosing a motion cluster c_j in the source domain, apart from the vehicle pose, another important consideration is the size of the training set used for learning DPM_j. In general, training on a larger amount of data, leads to a better generalization. This is especially true when the learning procedure needs to infer latent variables. The Deformable Parts Model (DPM) [104] treats the positions of the object parts as latent variables and employs a latent SVM to infer them from the data; therefore a large training set is crucial for learning an accurate DPM model. Based on all these factors, given a cluster c_i in the target domain, we can choose a cluster c_j in the source domain $\mathcal{S}$ and transfer its object recognition model DPM_j for detecting vehicles in the source domain according to the following criterion:

$$\begin{aligned} \text{DPM}_j \quad &= \arg\min_{j \in \mathcal{S}} w_a ||\mathbf{A_i} - \mathbf{A_j}||_2 + w_z ||\mathbf{Z_i} - \mathbf{Z_j}||_2 \\ &\quad + w_s \left(1 - \frac{|S_j|}{|S_{max}|}\right) \end{aligned} \tag{5.3}$$

where w_a, w_z and w_s are the relative weights assigned to the difference in the azimuthal direction $\mathbf{A}$, the difference in the motion direction $\mathbf{Z}$ and the relative size of the training dataset $|S|$ corresponding to cluster c_j. These weights are chosen by cross-validation. $|S_j|$ is the cardinality of the training set of cluster c_j and ($|S_{max}| = 20000$) is the cardinality of the largest cluster. The third term can be thought of as a penalty term which attempts to avoid selecting DPM models trained on small amounts of data by penalizing them. While our approach is exceedingly simple, our experiments demonstrate that given a large and diverse set of source

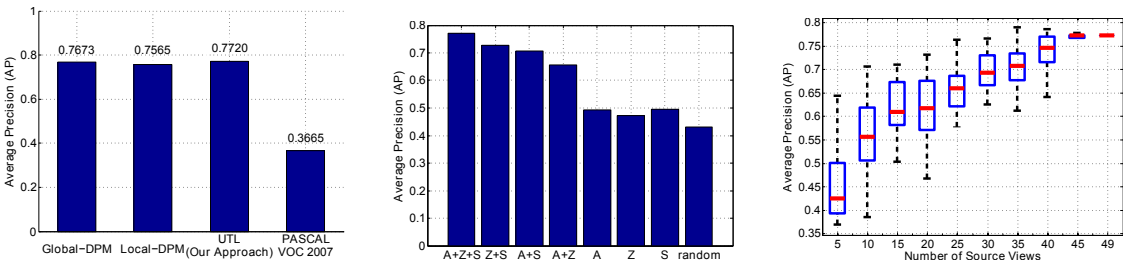
domains $\mathcal{S}$, our approach can outperform a DPM based object detector that utilizes labeled data from the target domain.

5.4 Experiments and Results

In order to evaluate our approach, we collected a test dataset consisting of about 3000 images collected from the same set of 50 cameras that were used for collecting the training data. From each camera viewpoint, images were collected at different times of the day and contain large variations in illumination due to the changes in the direction of sunlight and the resulting reflections and shadows from buildings. Apart from the viewpoint which changes significantly across the cameras, the amount of traffic also varies. On an average each test image contains between two and three vehicles.

For the purpose of evaluating our unsupervised transfer learning approach, we adopt a leave-one-out scheme, where each stage involves treating a particular camera viewpoint as the target domain and the remaining cameras as the source domains. Hence, during the evaluation of a test image from a target domain, none of the training images collected from that camera viewpoint are used for learning any of the object recognition models that might be transferred from the source domain. Given a target camera viewpoint, the most appropriate object detection models are chosen from the source domain according to the distance criterion (Equation (5.3)); we refer to this approach as Unsupervised Transfer Learning (UTL).

We follow the same experimental protocol that was used in the PASCAL VOC



(a) Supervised learning vs. unsupervised transfer learning

(b) Performance of different combinations of the distance measure

(c) Performance variation with amount of source-domain data

Figure 5.5: Performance of our Unsupervised Transfer Learning (UTL) approach.

2006 challenge - a predicted bounding box is considered to be correct if its overlap with a ground-truth bounding box is more that 50%, otherwise it is considered a false positive. Multiple detections of the same ground-truth object are penalized. Different models are compared based on the Average Precision (AP) of their precision-recall curve on the test set.

5.4.1 Comparison to Target-Domain Models

We compare the performance of our approach (UTL) against three different methods - *Local-DPM* and *Global-DPM* - that utilize training data from the target domain, and a DPM based model trained on the PASCAL VOC 2007 training set which does not utilize data from the target domain. In *Local-DPM*, for each camera viewpoint, we build DPM models consisting of two components, for each motion cluster in the viewpoint. These models are then evaluated on test images captured from the same viewpoint. The *Local-DPM* method represents the performance of the DPM object recognition model which has access to training data from the target

domain. In the case of *Global-DPM*, we utilize all the training images from each camera viewpoint to learn a DPM based object recognition model. The number of components in *Global-DPM* was set to eight as it resulted in the best performance. The *Global-DPM* approach, in addition to training data from the target domain, also utilizes training data from all the other source domains. The results are shown in Figure (5.5a). We can see that our approach, *UTL*, which is an unsupervised method(w.r.t. the target domain) performs even better than *Local-DPM* and *Global-DPM*, which have utilized labeled training data from the target domain. While it may seem surprising that our approach can outperform *Local-DPM*, which is based on the same object recognition model and also has access to training data from the target domain, the size of the local training dataset plays an important role. In some cases a model trained on a slightly different viewpoint but with a larger amount of training data can outperform a model trained on the same viewpoint. At the other extreme, simply learning a model from the entire training data might also be suboptimal as indicated by the performance of *Global-DPM*, which is slightly less than *UTL* despite the fact that it utilizes the entire data for training. We conjecture that *Global-DPM* is disadvantaged by its grouping of the components based on the aspect ratio of the training images instead of a more semantic criterion(e.g. the camera-viewpoint/motion-cluster hierarchy used by us), a point that was also made in [98].

UTL also offers a significant speedup over view-invariant methods which attempt to learn appearance models of all viewpoints simultaneously, such as *Global-DPM* or the discriminative mixture-of-templates [98]. *UTL* selects a two component

local DPM model corresponding to each motion cluster in a viewpoint. Each camera viewpoint contains two motion clusters on average, hence *UTL* requires evaluation of four DPM components resulting in a speedup by a factor of two over *Global-DPM*, which consists of an eight component DPM model.

We also compare our approach to a DPM model trained on the *car* class of the PASCAL VOC 2007 training set [111]. The PASCAL VOC training images contain four different orientations *frontal*, *rear*, *left* and *right* and we found the model trained on VOC 2007 to be the best performing among VOC 2006-2009. However, its performance was substantially poor compared to models learnt using our training data (Figure 5.5a). While this is not a completely fair comparison, it demonstrates that training on high quality in-domain data can have a significant impact on performance, and that our approach offers an effective mechanism for transferring information from a closely related source domain to the target domain. Moreover, it also reflects positively on the quantity and diversity of our training data and highlights the difficulty of our test set.

5.4.2 Distance Measure

For each motion cluster in a new camera viewpoint, our approach(*UTL*) utilizes a distance measure (Equation (5.3)) to identify the most appropriate DPM models from the source domain. The distance measure consists of three components: the difference between the Azimuthal direction (A) of the vehicles in the two clusters, the difference between the Zenith angle (Z) of the vehicles and the relative size (S)

of the training set of the motion cluster in the source domain. In order to demonstrate the importance of each of the three components that comprise our distance measure, we compare our approach *UTL* using all three components of the distance measure(A+Z+S) for selecting the most appropriate models, against all possible combinations of the individual components of the distance measure. The results are shown in Figure (5.5b) and they indicate that using all three components together significantly outperforms all other combinations using just one or two components, confirming the significance of each of the components that comprise our distance measure. We also evaluate the accuracy when the DPM models corresponding to each motion cluster are randomly chosen from the source domain, which results in the lowest accuracy.

5.4.3 Amount of Source-Domain Data

To study how the amount of source-domain data affects our approach, we evaluate the performance of our method by transferring recognition models for a given target domain from a subset of k randomly chosen source domains (camera viewpoints). Figure 5.5c plots the performance of our approach as a function of the number of source views(k). The largest value of k is 49, which corresponds to the case when all the camera viewpoints other than the target camera are treated as source domains. When $k < 49$, the performance is the mean over fifty runs, with a set of source domains of size k being randomly chosen during each run. It can be seen that the performance of our unsupervised transfer learning method increases

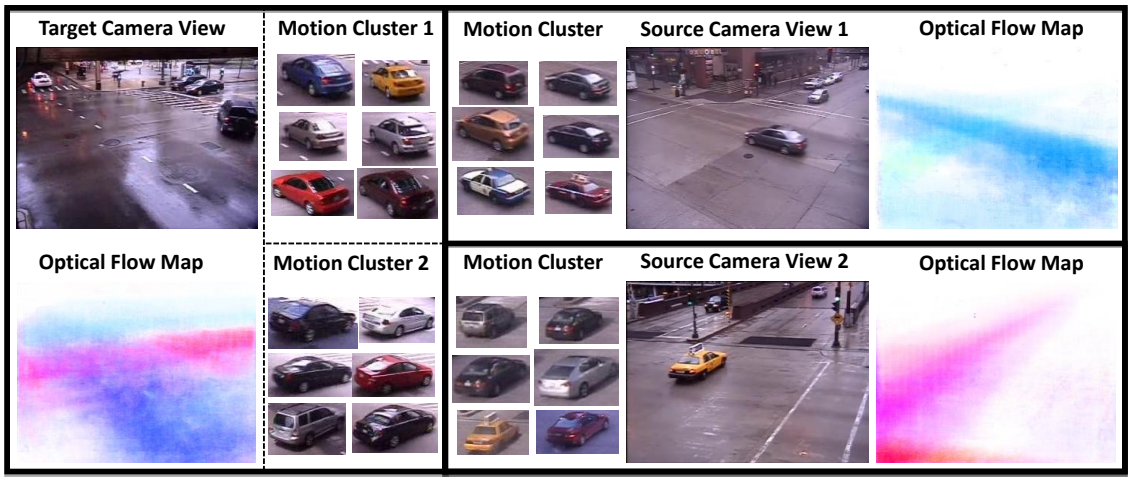


Figure 5.6: Top left contains a camera view in the target domain along with the optical flow map of the scene, which shows vehicles moving in two different directions, and examples of images of vehicles from the two motion clusters. The top and bottom right show the camera viewpoints from the source domain that were selected for transferring the object recognition model along with sample images of vehicles from the training set and the optical flow map of the specific motion-cluster. Note the similarity between the poses of the vehicles in the target and the source motion clusters.

with an increase in the number of training camera viewpoints and asymptotically approaches and even surpasses the supervised upper bound represented by the *Local-DPM* and the *Global-DPM* methods. This is expected, as a larger number of camera viewpoints implies a higher probability of there existing camera viewpoints in the source domain containing vehicles in poses that closely match poses of vehicles in the target domain.

5.4.4 Qualitative Analysis

Figure 5.6 is an illustration of our approach for a sample camera viewpoint. Given a new camera-view consisting of two motion-clusters corresponding to vehicles moving in two different directions, our approach first selects motion-clusters from viewpoints in the source domain where vehicles are expected to be present in similar poses and utilizes the detectors trained on them for vehicle detection in the new view. Given the large number of different camera-viewpoints present in the source domain, we are able find motion-clusters in the source domain with relatively similar poses. Consequently the object recognition models trained on them are able to perform well in the target domain. Some detection results on images captured from different camera viewpoints are shown in Figure (5.7).

5.5 Conclusion

We have presented an approach for view-invariant vehicle detection in traffic surveillance videos, which learns a large number of view-specific detectors during the training phase and given an unseen viewpoint exploits scene geometry and vehicle motion patterns to select a particular view-specific detector for object detection. The key advantage of our approach is that it enables utilization of fast and simple view-specific object detectors for accurate view-invariant object detection. Although we have demonstrated the effectiveness of our approach on the task of vehicle detection, our approach can be potentially applied to other object detection problems where the object pose can be inferred using auxiliary information, in order to improve

Figure 5.7: A few examples of vehicle detection on images captured from different traffic surveillance cameras. Notice the extreme variations in viewpoint, scale and illumination. Also, note that we do not detect vehicles below a certain scale and hence miss some small vehicles.

speed and accuracy.

Chapter 6

Combining Multiple Kernels for Efficient Object Recognition

6.1 Introduction

We address the problem of combining multiple heterogenous features for image classification. Categorizing images based on stylistic variations such as scene content and painting genre requires a rich feature repertoire. Classification is accomplished by comparing distributions of features, e.g., color, texture, gradient histograms [35, 59, 36]. For instance, Grauman and Darrell proposed the Pyramid Match Kernel (PMK) to compute Mercer kernels between feature distributions for Support Vector Machine (SVM) based classification. This has been shown to be effective for object categorization [35] and scene analysis [36]. Approaches such as PMK would compute a kernel matrix for each feature distribution. We explore techniques for combining the kernels from multiple features for efficient and robust recognition.

A number of techniques have been proposed to learn the optimal combination of a set of kernels for SVM-based classification. Lanckriet et al. proposed an approach for Multiple Kernel Learning (MKL) through semi-definite programming [43]. Sonnenburg et al. generalized MKL to regression and one-class SVMs, and enhanced the ability to handle large scale problems. Rakotomamonjy et al. increased the efficiency of MKL and demonstrated its utility on several standard datasets including the UCI repository [37]. They compute multiple kernels by vary-

ing the parameters of polynomial and Gaussian kernels, and apply MKL to compute an optimal combination. Bosch et al. learn the optimal mixture between two kernels - shape and appearance - using a validation set [60]. Varma and Ray propose to minimize the number of kernels involved in the final classification by including the L_1 norms of the kernel weights in the SVM optimization function [61]. Bi et al. proposed a boosting-based classifier that combines multiple kernel matrices for regression and classification [62].

The efficiency of MKL-based SVM classifiers during the testing phase depends upon the number of support vectors and the number of features. In general, multi-class problems requiring subtle distinctions entail a large number of support vectors. The computational cost is substantial when the kernels are complex, e.g., matching similarity of feature distributions. Is it possible to reduce the number of complex kernel computations while maintaining performance? We propose an approach for combining multiple kernels through a feature selection process followed by SVM learning. Let $K_m(.,.)$ be the kernel values for the m^{th} feature channel computed using approaches such as the Pyramid Match Kernel. The columns of K_m are considered to be features embedding the images in a high-dimensional space based on similarity to training examples. During the training phase, a subset of the columns are chosen using Gentle Boost [57] based on their discriminative power, and a new kernel K is constructed. This is provided as input to an SVM for final classification. Kernels of test images need to be computed for only the chosen set of columns - much smaller than the full set of kernel values. This results in substantial reductions in computational complexity during the testing phase. The consequent approach is

simple and relies on well understood techniques of Boosting and SVMs. Boosting methods have previously been used for feature selection [45], to learn kernels directly from data [47, 48], and for selecting a subset of kernels for concept detection in [49].

We compare our Boosted Kernel SVM (BK-SVM) approach with the Efficient Multiple Kernel Learning (EMKL) approach proposed in [37]. EMKL has been shown to increase the efficiency of kernel learning while enabling the use of a large number of kernels within SVM. It uses all the kernel values for classification - a superset of the features obtained by the greedy Boosting-based selection. BK-SVM and EMKL are tested in two scenarios: standard datasets from the UCI repository [58] and a novel Painting dataset. Results indicate that BK-SVM's classification accuracy is comparable to that of EMKL, with the additional advantage of a much smaller number of complex kernel computations.

Currently, paintings are being extensively digitized in order to preserve them and make them more widely accessible. Digital collections of paintings play an important role in preserving our cultural heritage. Automatic indexing and annotation of such painting collections according to style, artist or period would considerably reduce the manual effort required for such tasks. Supporting query and retrieval on such collections over the internet would make many rare paintings more widely accessible. In this work, we apply our BK-SVM method to the task of annotation of paintings according to their genre, which could be applied to indexing as well as query and retrieval from painting collections.

The Painting dataset consists of nearly 500 images downloaded from the Internet - the task being to classify images into 6 genres. This provides a good testbed as

the classification is subtle, requiring a large variety of features. Recently, there have been studies on the classification of paintings based on their style, artist, period and brushwork [31, 32, 38, 42, 39]. A semi-supervised method employing a variety of feature channels to annotate painting brushwork was presented in [31]. In [32] paintings are classified according to artist. Li et al. [42] have used 2D multi-resolution HMMs with multi-level Daubechies wavelet coefficient features to identify the artists of ancient Chinese paintings. In [64], high level semantic concepts are combined with low level image features to annotate paintings based on period, style and artist. In some of these methods such as [31, 64] a high level of domain knowledge has been used to develop the hierarchy of classes and to select appropriate image features. We use a large repertoire of simple features and rely on machine learning to obtain the combination best suited for the classification. This provides the potential for application in other categorization tasks.

The next section presents details of combining multiple kernels, followed by experiments on the UCI datasets. Section 6.4 presents the Painting dataset, the features used and the experimental results.

6.2 Learning a Mixture of Kernels

Content-based image categorization typically represents images with histograms or distributions of features from channels such as texture, color and local gradients [33, 41]. Classification is performed by comparing such distributions. Grauman and Darrell [35] proposed the Pyramid Match Kernel (PMK) for efficiently comput-

ing Mercer kernels between feature distributions and apply it to SVM based object categorization [35]. A closely related approach used spatial distributions of features for scene recognition [36]. These techniques use SVM to learn the manifold of image categories and show good generalization. However, classifying images based on subtle style variations, e.g., painting genres, requires a large repertoire of feature channels. Techniques such as PMK would compute a kernel matrix for each feature channel. We are thus faced with the problem of determining the best mixture of the kernels for a given classification task.

A number of Multiple Kernel Learning (MKL) techniques have been proposed to compute linear combinations of kernels for classification by SVM [43, 37, 44]. Let $\{K_1, K_2, \ldots, K_M\}$ be the kernel matrices computed for various feature modalities. MKL computes an optimal classification kernel

$$K(q_i, q_j) = \sum_{m=1}^{M} \beta_m K_m(q_i, q_j) \tag{6.1}$$

where $\{q_1, q_2, \ldots, q_N\}$ are the training images and β_m is the weight assigned to kernel K_m. Recent MKL techniques have progressively improved training efficiency, e.g., [37, 60]. However, classifying a test image x, using a non-linear SVM, requires computing its kernel value with respect to the selected set of training support vectors S for all feature channels with $\beta_m \neq 0$, i.e. $K_m(q, x)\ \forall q \in S$ and $\forall m$ where $\beta_m \neq 0$. This has $O(c\tilde{N}\tilde{M})$ computational complexity where

- c is the complexity of computing the kernels. This is significant when com-

puting the similarity of distributions.

- $\tilde{N}$ is the number of support vectors, which is less than or equal to the size of the training set, N. Classification problems with difficult decision boundaries require a large set of support vectors. Some approaches propose to reduce this by approximating S with a reduced set of vectors, e.g., [55]. However, they are unsuitable for our case as each kernel is constructed from a different feature modality. Moreover, it is desirable to include as many training images as possible for good generalization (large N).

- $\tilde{M} = |\{m|\beta_m \neq 0\}|$. MKL methods reduce $\tilde{M}$ by imposing sparsity constraints on the weights β_m [44]. However, this may not provide significant benefits when a large variety of features are required for classification.

Is it possible to reduce the number of kernel computations while maintaining performance?

Consider a vector constructed for a test image by concatenating its kernel values with all the training images. For an image x, this would be an NM dimensional vector

$$\mathbf{f}(x) = \langle K_1(q_1, x) \dots K_1(q_N, x) \dots K_M(q_1, x) \dots K_M(q_N, x) \rangle \tag{6.2}$$

We use Gentle Boost to determine the set P, containing the most discriminative dimensions of $\mathbf{f}(\mathbf{x})$, for the classification problem. The size of P is chosen such that $|P| \ll \tilde{N}\tilde{M}$. This results in a reduced dimensional vector for each image,

denoted by $\tilde{\mathbf{f}}(x)$. An SVM is trained to classify images based on the $\tilde{\mathbf{f}}$'s. E.g., for a linear SVM, the kernel between two images x and y would be

$$\Phi(x,y) = \sum_{\langle n,m\rangle \in P} K_m(x,q_n) K_m(q_n,y) \tag{6.3}$$

For each test image, this requires $O(|P|)$ complex kernel $K_m(.,.)$ computations, and $O(N|P|)$ computations of a simpler kernel such as linear or RBF. This significantly reduces the computational complexity.

To better understand the nature of $\Phi(.,.)$, notice that the Pyramid Match Kernel between two images x and y can be abstracted as a dot-product between two bit-vectors,

$\psi_m(x)^T \psi_m(y)$, where m is the feature channel [35]. Therefore, eq.(6.3) is equivalent to

$$\begin{aligned} \Phi(x,y) &= \sum_{\langle n,m\rangle \in P} \psi_m(x)^T \psi_m(q_n) \psi_m(q_n)^T \psi_m(y) \\ &= \sum_m \psi_m(x)^T \left[\sum_{\langle n,m\rangle \in P} \psi_m(q_n) \psi_m(q_n)^T \right] \psi_m(y) \end{aligned} \tag{6.4}$$

The inner matrix, $A_m = \sum \psi_m(q_n)\psi_m(q_n)^T$, is a semi-definite matrix. It is easy to show that for a RBF SVM

$$\Phi(x,y) = \exp \frac{1}{\sigma^2} \sum_m \|\psi_m(x) - \psi_m(y))\|^2_{A_m} \tag{6.5}$$

Intuitively, A's are akin to covariance matrices of the exemplar images in P, the important difference being that $\psi_m(q_n)$ are not zero mean. When P is constructed to maximize discrimination between classes, A defines a discriminative projection.

We note that the approach does not restrict the number of support vectors chosen by the SVM. It only restricts the SVM's kernel to be based on a limited number of base kernel columns.

6.2.1 Boosting for Feature Selection

Discriminative feature selection is a well studied problem in machine learning, e.g., Xiao et al. propose a variant of boosting called Joint Boost for feature selection [45]. We use Gentle Boost for its simplicity and robustness [57, 56]. Let $\mathbf{f}$'s be d dimensional vectors. d is typically large; in our case $d = NM$. The basic version of Gentle Boost defines a set of weak learners $h(\mathbf{f})$ where each $h(.)$ is a linear classifier along a single dimension. The algorithm iteratively chooses a set of weak learners to maximize classification accuracy. The weak learner chosen at the t^{th} iteration, namely $h_t(.)$, is the one providing maximal increase in classification accuracy with respect to the set of previously chosen classifiers $h_1, \ldots, h_{t-1}$. Thus, the choice of dimensions depends upon the incremental benefit relative to previous choices. Inspite of the greedy nature of the selection process, Boosting has been shown to perform well in many classification tasks [56]. The outline of Gentle Boost is given below:

- Given: $(x_1, y_1), \ldots, (x_n, y_n)$ where $x_i \in X$ and $y_i \in \{-1, 1\}$
- Initialize the weights corresponding to the training samples $W(i) = \frac{1}{n}$

- For $t = 1, \ldots, T$
  - Choose confidence value $\alpha_t \in R$
  - Find the classifier h_t which minimizes the classification error with respect to the distribution W_t
  - Update the weights $W_{t+1}(i) = \frac{W_t(i)e^{-\alpha_t y_i h_t(x_i)}}{Z_t}$

    where Z_t is a normalization factor.
- $\{h_t\}$ are the selected features.

Table 6.1: Experiments on UCI Dataset

Dataset			**BK-SVM**		**EMKL**	
name	size	kernels	accuracy	comp.	accuracy	comp.
Liver	345	91	66.2 ± 4.7	40	65.0 ± 2.3	1607 ± 324
Ionosphere	351	442	92.1 ± 3.6	40	92.3 ± 1.4	1496 ± 266
Pima	768	117	73.7 ± 6.4	60	75.8 ± 1.6	3123 ± 526
Sonar	208	793	76.3 ± 4.9	20	78.6 ± 4.2	2538 ± 351

6.3 Experiments with UCI Datasets

The boosting-based feature selection is an efficient but greedy approach. To observe its performance penalties, BK-SVM was applied to four datasets from the UCI repository, specifically the Liver, Ionosphere, Pima and Sonar datasets. The

kernels were simple polynomial and Gaussian functions. Here, the motivation was solely to empirically observe the performance on standard datasets. The efficiency gains become evident for more complicated kernel functions used later in the Painting datasets.

The classification results were compared with those of the Efficient Multiple Kernel Learning (EMKL) algorithm described in [37]. For each dataset, a large number of Gaussian and polynomial kernels are computed as described in [37]. The base kernels include Gaussian kernels with 10 different bandwidths σ on all variables and on each single variable, and also polynomial kernels of degree 1 to 3. EMKL and BK-SVM are used to learn a mixture of the kernels appropriate for classification. During the testing phase, the number of kernel computations required in EMKL is a product of the number of kernels selected and the number of support vectors. While in BK-SVM, this depends upon the number of kernel columns chosen by boosting.

The classification results are summarized in Table 6.1. They indicate that BK-SVM performs close to the baseline EMKL approach even though the number of kernel computations is more than an order of magnitude lower. The loss of performance of approximately 2% may be ascribed to the greedy selection of kernel columns. The results also demonstrate the scalability of our method, which performs comparably to EMKL even in the case of the Sonar dataset where a large number of kernels(793) are used for learning with only a small number of training examples(104). These trends are reflected in the experiments with the painting dataset - described in the next section. The modest loss in performance is outweighed by the large decrease in computational complexity.

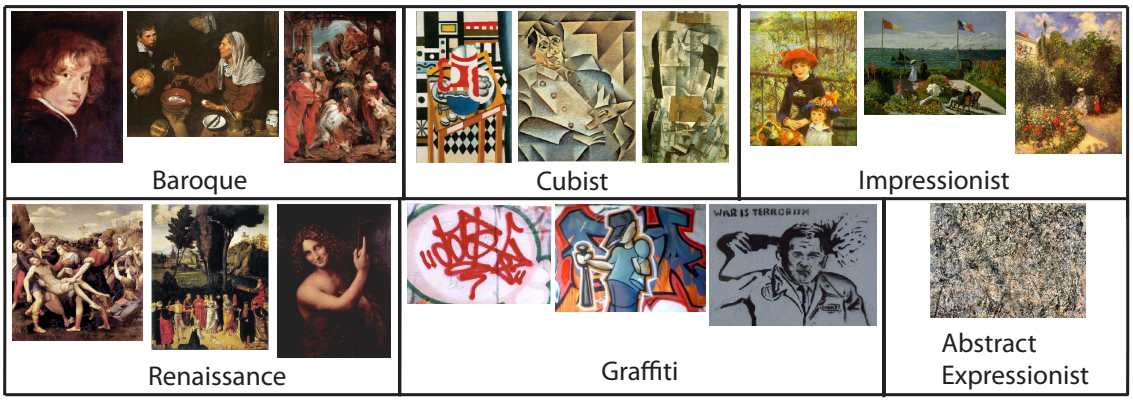


Figure 6.1: Example images from the Painting database

6.4 Painting Dataset

BK-SVM was applied to painting genre classification. A dataset of 81 Abstract Expressionist, 84 Baroque, 84 Cubist, 82 Graffiti, 89 Impressionist and 78 Renaissance (total of 498) paintings was collected from the Internet. The painting styles along with the painters of each style are listed in Table 6.2. Some of the public domain images are shown in Figure 6.1. The distinguishing features for painting styles are not clearly defined due to its abstract nature. There is high intra-class variation due to differences between the painters of a particular style and also between the different paintings of individual painters [63]. The content in the paintings varies significantly and occasionally paintings of different styles depict the same scene, further complicating the problem. Having been compiled from a variety of sources, the images have variations in scale and illumination as well. The classification task is complex, requiring a rich set of features, making this a good testbed for BK-SVM.

6.4.1 Features

Inspired by previous studies on painting classification, a large variety of features are computed. Each feature channel produces a distribution of filter responses for a given image. The similarity of images is defined as the match between the distributions.

6.4.1.1 Texture

Texture features capture brushwork and characteristics of the depicted scene. They have been shown to be effective in classification of paintings [42, 31, 38]. We employ the MR8 filter bank [30] as it responds to both isotropic and anisotropic textures and was observed to perform better than Gabor filter banks. The MR8 filter bank consists of a Gaussian and a Laplacian of Gaussian with $\sigma = 10$ and oriented edge and bar filters at 3 scales $(\sigma_x, \sigma_y) = \{(1, 3), (2, 6), (4, 12)\}$ and 6 orientations. Only the maximum response is recorded at each scale for each of the edge and bar filters across all orientations. The responses at all the pixels are combined to form a set of vectors, denoted by F_{texture}.

6.4.1.2 Histograms of Oriented Gradients (HOG)

HOG based descriptors have been extensively used for representing local shape [33, 40, 41]. They have some degree of invariance to illumination and geometric transformations. We compute two types of features using HOG:

1. F_{HOGdense}: set of HOG features on overlapping 8×8 sized patches placed on

a dense regular grid with a spacing of 4 pixels - similar to [33].

2. $F_{\text{HOGsparse}}$: sparse set of HOG features computed on 8×8 patches centered on all edge points. This was inspired by [40].

6.4.1.3 Color

Color features have been previously employed for classifying paintings [38, 39]. We use local histograms to represent color features consisting of 10 bins of the pixel intensities of each color channel. The histograms are computed in 8×8 sized patches centered on a dense grid over the image. This generates a set of vectors denoted by F_{color}. The histograms of different color channels were concatenated because the joint histograms were quite sparse. Experiments indicated that RGB, HSV and LUV had similar performance. Only results for RGB color-space are presented here.

6.4.1.4 Saliency

Edge Continuity is used to enhance the saliency of long continuous curves relative to scattered and cluttered edges. We use the technique described in [34] for computing the saliency maps of the images. HOG features are extracted from these saliency maps from patches centered on edges having high saliency. The obtained set of HOG vectors is denoted by F_{HOGsal}.

(a) Original Image (a) Gradient Magnitude Map (a) Saliency Map

Figure 6.2: Salient Edges

6.4.2 Pyramid Match Kernel

Each of the features produces a set of vectors for a given image. For each feature channel, similarity between images is computed based on the similarity between the two sets of vectors, computed using Pyramid Match Kernel (PMK) [35]. The sets can have different cardinalities. The approach has been shown to be efficient and effective for image classification. In this section we briefly describe the kernel. Let X and Y be two sets of feature vectors in a d-dimensional feature space. Now consider $L+1$ levels of histograms $H^0, H^1, \ldots, H^L$. Level 0 of the histogram consists of just 1 bin which is the entire space, level 1 of the histogram consists of 2^d bins equally dividing the feature space into two parts along all dimensions. Similarly level l of the histogram consists of $D = 2^{dl}$ bins. Let H_X^l and H_Y^l denote the histograms of X and Y at level l with $H_X^l(i)$ and $H_Y^l(i)$ being the number of feature vectors of X and Y respectively falling into the ith bin at level l. A histogram intersection gives the number of matches at this level.

$$I(H_X^l, H_Y^l) = \sum_{i=1}^{D} \min(H_X^l(i), H_Y^l(i))$$

But note that all the matches at level $l+1$ are also matches at this level and

hence the number of new matches at level l is $I(H_X^l, H_Y^l) - I(H_X^{l+1}, H_Y^{l+1})$. The matches at level l are weighted by $\frac{1}{2^{L-l}}$ in order to give higher weights to matches which happen at smaller bin sizes and hence have a higher similarity. The total match between X and Y at all levels is defined as the similarity between X and Y

$$K(X,Y) = I(H_X^L, H_Y^L) + \sum_{l=0}^{L-1} \frac{1}{2^{L-l}} I(H_X^l, H_Y^l) - I(H_X^{l+1}, H_Y^{l+1})$$

To avoid biasing the kernel toward larger input sets it is normalized

$$K(X,Y) = \frac{K(X,Y)}{\sqrt{K(X,X)K(Y,Y)}}$$

This normalization also ensures that $\forall X, Y\ K(X,Y) \in [0,1]$. It has been shown that Pyramid Match produces a Mercer kernel and can be directly used in an SVM.

6.4.3 Classification Results

Training the BK-SVM consists of the following steps:

- Each of the M described features is extracted for all training images q_i.
- PMK was used to compute kernel values $K_m(q_i, q_j)$, $\forall q_i, q_j, m$, producing M kernel matrices, $K_1, \ldots, K_M$.
- A vector $\mathbf{f}_i$ is constructed for each q_i by concatenating the kernel values as defined in eq.(6.2).

- Boosting is used to select a set of L dimensions that best classify $\mathbf{f}_i$'s into the painting genres. The number of exemplar images selected is equal to the number of iterations of boosting and thus can be easily controlled.

- A new RBF kernel matrix Φ is constructed from the selected dimensions (i.e. columns of K_m's) through the relation in eq.(6.5). A one-vs-all multi-class SVM is trained on Φ.

During the testing phase, PMK is computed between a given test image and the L selected training images. Classification is performed through the trained RBF SVM.

For comparison, EMKL was employed for the same classification task. For EMKL, we learn separate kernels for each individual classifier, using the same parameters that were used for the UCI datasets($C = 100$,maximal number of iterations=500,duality gap=0.01). All the experiments were repeated 10 times with a 5-fold cross-validation.

6.4.3.1 Individual Features

Table 6.3 shows the performance of the individual classifiers, only the net results are shown due to space constraints. We now discuss the performance of individual features.

Color: In Baroque and Renaissance paintings, darker colors are used and this makes color histograms particularly useful for discriminating them from the other classes. With color features alone, Baroque paintings had a recognition rate of 91%. Color

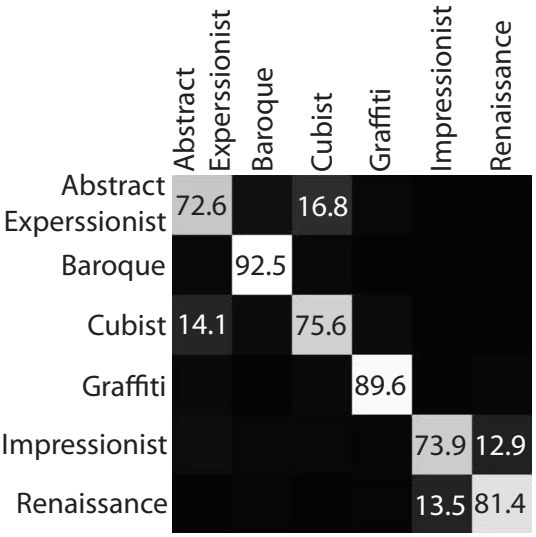


Figure 6.3: Confusion Matrix for the painting dataset

features are also useful for identifying Impressionist paintings as they tend to depict outdoor scenes with sunlight, landscapes and greenery.

Texture: The texture feature proved useful for distinguishing Impressionist images as they have distinctive brush strokes. Baroque paintings being darker, generate low responses with the filter banks and are also easily identified.

HOG: The cubist paintings are composed of dense geometrical structures such as straight lines, cubes and cylinders. Consequently, local shape features such as the dense HOG are useful in distinguishing them. The sparse HOG features encode the local shape around the edge points and prove useful for identifying Impressionist paintings.

Saliency: Graffiti paintings tend to have smooth continuous contours, which get enhanced in the saliency maps(Fig. 6.2), computed using edge continuity techniques[34]. The local shape features around these salient contours help discriminate them.

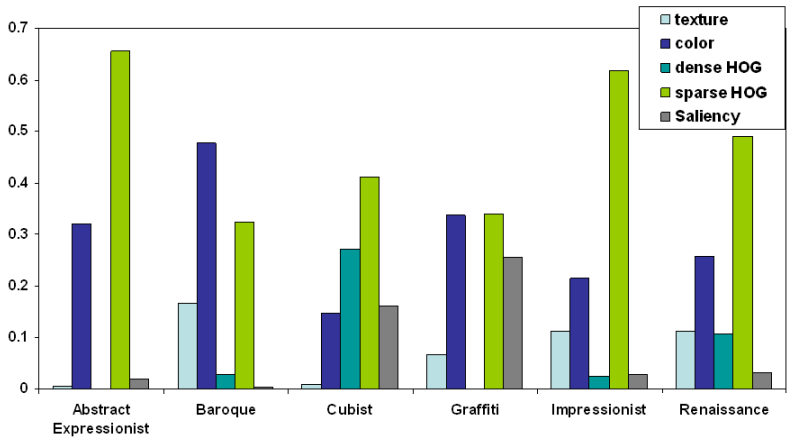


Figure 6.4: Avg. kernel weights learnt by EMKL for each classifier

These saliency based features help recognize Graffiti paintings with an accuracy of 82%.

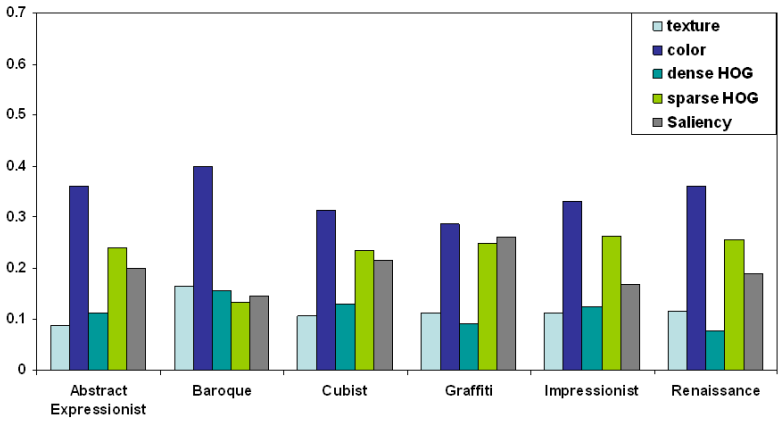


Figure 6.5: Avg. proportion of exemplar images selected from the feature channels for each classifier

6.4.3.2 Combination of Features

The features, in general, perform quite well individually and also complement each other resulting in a significant improvement in performance when combined.

For instance, on the sole basis of color, a dark colored graffiti painting may be confused as a baroque painting. However, local shape information provided by saliency maps helps reduce this confusion. The results are listed in Table 6.3. They indicate that both the EMKL and our method perform much better than each of the individual feature channels. The confusion matrix obtained after combining features using BK-SVM is shown in Fig. 6.3. There is some degree of confusion between abstract expressionist and cubist paintings and most of the misclassifications happened to be abstract expressionist paintings containing geometrical structures characteristic to cubist paintings or cubist paintings lacking these geometrical shapes. There are also some errors between impressionist and renaissance paintings.

6.4.3.3 Feature Selection

To gain further insight into the construction of the individual one-vs-all classifiers, we looked at the average weights allocated by EMKL to the kernels for each individual classifier(Fig. 6.4). Color being an important feature was assigned a high weight in each of the individual classifiers and as expected, it turned out to be the most dominant feature for distinguishing Baroque paintings. Similarly the saliency kernel is weighted relatively high in the Cubist and Graffiti classifiers. Texture is also important in case of the Baroque, Impressionist and Renaissance classes. Sparse HOG features are assigned high weights in all the classifiers, indicating the significance of local shape information. Dense HOG features are allocated high weights in the Cubist classifier as expected. On the whole, the weights seemed quite

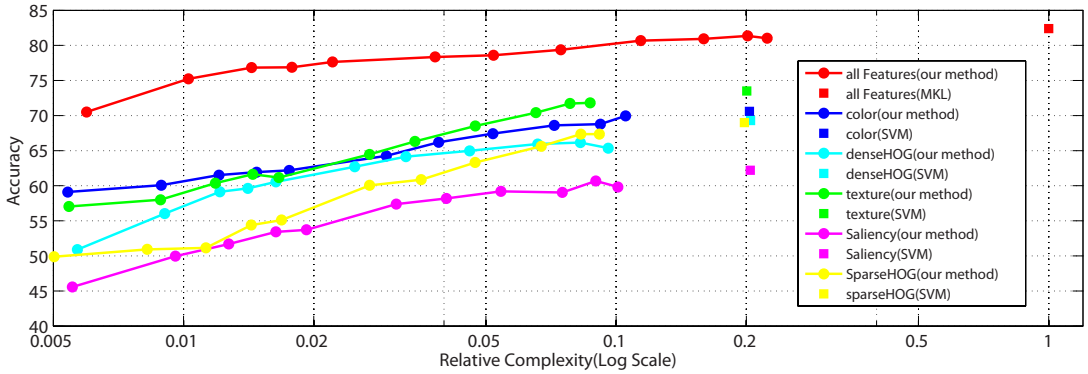


Figure 6.6: Variation in performance as a function of the number of features selected for the painting dataset

intuitive with features that distinguished a particular class well, being assigned a higher weight in the respective classifier. However, texture was weighted relatively low which is surprising, given the fact that it performs quite well individually. We conjecture that since both texture and HOG are based on local edges, they contain redundant information resulting in texture being ignored.

We did a similar study for BK-SVM, where we examined the proportion of exemplar images selected from each kernel for the individual classifiers(Fig. 6.5). Though some of the above mentioned trends were observed, like color and saliency being important for the baroque and graffiti paintings respectively, no single feature dominated the individual classifiers. We hypothesize that this is a result of the lack of any external constraints imposed by our method unlike the sparsity constraint imposed by EMKL.

6.4.3.4 Efficiency

Figure 6.6 plots the performance of our method as a function of the number of kernel computations required. It can be seen that BK-SVM reduces the number of kernel computations required by a factor of 10, while suffering only a minor (1-2%) reduction in accuracy. It can also be observed that the performance decreases gradually as the number of features selected is decreased. At a relative speedup of 100 with respect to MKL, BK-SVM is still better than each of the individual feature channels and only 7% less accurate than MKL.

We also apply our method on the individual kernels and compare the performance of BK-SVM with that of a SVM using a single kernel. Here BK-SVM is used to learn a kernel from a subset of the training images, while SVM uses the kernel computed from the entire training set. As expected, the performance increases with the increasing number of features selected and approaches that of a SVM while being more efficient. Figure 6.6 once again underscores the importance of combing multiple features for improving accuracy both for EMKL as well as BK-SVM.

In the Painting dataset, BK-SVM requires nearly 10 times fewer kernel computations than EMKL for achieving comparable accuracy. This speedup, though substantial, is less compared to the 40-120 time reduction achieved on the UCI datasets. There are two plausible explanations. Firstly, the painting dataset has multiple classes, which makes the decision boundaries more complex than in case of the UCI datasets, which have only two classes. Secondly, the UCI dataset experiments use base kernels produced by varying the parameters of Gaussian and

polynomial kernels, many of which are likely to be redundant. Hence, a sparse set of features selected by Boosting is sufficient to accurately approximate the optimal kernel. In case of the painting dataset, each of the base kernels are computed from different feature channels containing complementary information. Consequently, a number of exemplar instances are selected from each base kernel.

6.5 Summary

We have presented a simple and efficient approach for learning a mixture of kernels. Our method, which learns a mixture of kernels by greedily selecting exemplar data instances corresponding to each kernel using AdaBoost, has been shown to compare well to multiple kernel learning methods, while simultaneously reducing the number of kernel similarity computations required. The effectiveness of our method with respect to MKL has been demonstrated on some of the benchmark UCI datasets. We have also tested our method on an extremely diverse and challenging painting dataset, where a single feature channel is inadequate for classification. We combine multiple kernels computed from different feature channels, obtaining results comparable to the MKL method. The results provide evidence that our method is almost as accurate as the multiple kernel learning method, while being computationally much more efficient.

Table 6.2: Painting Classes

Painting Style	**Artist**
Abstract Expressionist	Arshile Gorky, Helen Frankenthaler, James Brooks, Jane Frank, Jean Paul Riopelle, Kenzo Okada, Paul Jenkins
Baroque	Anthony Van Dyck, Artemisia Gentileschi, Caravaggio, Diego Velazquez, Jan Vermeer, Nicholas Poussin, Peter Paul Rubens, Rembrandt
Cubist	Fernand Leger, Georges Braque, Gino Severini, Jacques Villon, Juan Gris, Lyonel Feininger, Pablo Picasso
Graffiti	-
Impressionist	Alfred Sisley, Camille Pissarro, Claude Monet, Frederic Bazille, Mary Cassatt, Pierre Auguste Renoir, Edouard Manet
Renaissance	Correggio, Raphael, Leonardo Da Vinci, Titian, Sandro Botticelli, Giorgione, Pieter Brueghel, Michelangelo

Table 6.3: Painting Classification Results

Feature	**Accuracy**
texture	73.5 ± 1.1
color	70.6 ± 1.1
dense HOG	69.3 ± 1.2
sparse HOG	69.0 ± 1.0
Saliency	62.2 ± 0.7
Combined EMKL	82.4 ± 0.9
Combined Our Method	81.3 ± 0.6

Bibliography

[1] B. C. Russell and A. Torralba and K. P. Murphy and W. T. Freeman, LabelMe: a Database and Web-based Tool for Image Annotation, IJCV 2008.

[2] A. Gupta and L. S. Davis, Beyond Nouns: Exploiting Prepositions and Comparative Adjectives for Learning Visual Classifiers, ECCV 2008.

[3] J. Deng and W. Dong and R. Socher and L.-J. Li and K. Li and L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database, CVPR 2008.

[4] A. Torralba and R. Fergus and W. T. Freeman, 80 million tiny images: a large dataset for non-parametric object and scene recognition, IEEE Transactions on PAMI, 2008.

[5] A. Sorokin and D. Forsyth, Utility data annotation with Amazon Mechanical Turk, Work. on Internet Vision, 2008.

[6] L. Ahn and L. Dabbish, Labeling Images with a Computer Game, ACM CHI 2004.

[7] S. Gould, R. Fulton, D. Koller, Decomposing a Scene into Geometric and Semantically Consistent Regions, ICCV 2009.

[8] J. Shotton and J. Winn and C. Rother and A. Criminisi, TextonBoost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context, IJCV 2007.

[9] E. Y. Chang, S. Tong, K. Goh, and C. Chang. Support vector machine concept dependent active learning for image retrieval. IEEE Transactions on Multimedia, 2005.

[10] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. In ICML, 2000.

[11] Y. Freund, H. S. Seung, E. Shamir, N. Tishby. Selective sampling using the query by committee algorithm. ML, 1997.

[12] D. MacKay. Information-based objective functions for active data selection. Neural Computation, 4(4), 1992

[13] N. Lawrence, M. Seeger, R. Herbrich. Fast sparse Gaussian Process method: Informative vector machines. NIPS, 2002

[14] X. Li, L. Wang, and E. Sung. Multilabel svm active learning for image classification. In ICIP, 2004.

[15] R. Yan, J. Yang, and A. Hauptmann. Automatically labeling video data using multi-class active learning. In ICCV, 2003.

[16] A. Holub, P. Perona, and M. Burl. Entropy-based active learning for object recognition. In CVPR workshop on Online Learning for Classification, 2008.

[17] Lindenbaum, M., Markovitch, S., Rusakov, D.: Selective Sampling for Nearest Neighbor Classifiers. ML (2004)

[18] Kapoor, A., Horvitz, E., Basu, S.: Selective Supervision: Guiding Supervised Learning with Decision-Theoretic Active Learning. In: IJCAI. (2007)

[19] Settles, B., Craven, M., Ray, S.: Multiple-Instance Active Learning. In: NIPS. (2008)

[20] P. Jain and A. Kapoor, Active Learning for Large Multi-class Problems, In CVPR 2009.

[21] S. Vijayanarasimhan and K. Grauman, Multi-Level Active Prediction of Useful Image Annotations for Recognition, NIPS 2008

[22] A. Kapoor, G. Hua, A. Akbarzadeh and S. Baker, Which Faces to Tag: Adding Prior Constraints into Active Learning, In ICCV 2009

[23] M. Galun, E. Sharon, R. Basri and A. Brandt, Texture segmentation by multiscale aggregation of filter responses and shape elements, ICCV, 2003.

[24] A. Rabinovich and T. Lange and J. Buhmann and S. Belongie, Model Order Selection and Cue Combination for Image Segmentation, In CVPR 2006

[25] Gökhan H. Bakir, Thomas Hofmann, Bernhard Schölkopf, Alexander J. Smola, Ben Taskar and S. V. N. Vishwanathan, Predicting Structured Data, MIT Press, 2007.

[26] I. Tsochantaridis, T. Hofmann, T. Joachims and Y. Altun, Support Vector Learning for Interdependent and Structured Output Spaces, In ICML, 2004.

[27] M. B. Blaschko and C. H. Lampert, Learning to Localize Objects with Structured Output Regression, In ECCV 2008.

[28] C. Desai, D. Ramanan and C. Fowlkes, Discriminative Models for Multi-Class Object Layout, In ICCV 2009.

[29] A. Kembhavi, B. Siddiquie, R. Miezianko, s. McCloskey and Larry S. Davis, Incremental Multiple Kernel Learning for Object Recognition, ICCV, 2009.

[30] M. Varma and A. Zisserman, Classifying Images of Materials: Achieving Viewpoint and Illumination Independence, ECCV, 2002.

[31] M. Yelizaveta, C. Tat-Seng and Ramesh Jain, Semi-supervised annotation of brushwork in paintings domain using serial combinations of multiple experts, ACM-MM, 2006.

[32] D. Keren, Recognizing image "style" and activities in video using local features and naive Bayes, Pattern Recognition Letters, 2003.

[33] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, CVPR, 2005.

[34] G. Guy and G. G. Medioni, Inferring Global Perceptual Contours from Local Features, IJCV 1996.

[35] K. Grauman and T. Darrell, The pyramid match kernel: discriminative classification with sets of image features, ICCV 2005.

[36] S. Lazebnik, C. Schmid and J. Ponce, Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, CVPR 2006.

[37] A. Rakotomamonjy, F. Bach and S. Canu and Y. Grandvalet, More efficiency in multiple kernel learning, ICML 2007.

[38] H.J. van den Herik and E. O. Postma, Discovering the Visual Signature of Painters, 2000.

[39] M. Yelizaveta, C. Tat-Seng and A. Irina, Analysis and Retrieval of Paintings Using Artistic Color Concepts, ICME 2005.

[40] S. Belongie, J. Malik, and J. Puzicha, Shape matching and object recognition using shape contextsIEEE Transactions on PAMI, 2002.

[41] D. Lowe, Distinctive image features from scale-invariant keypoints, IJCV 2003.

[42] J. Li and J. Z. Wang, Studying digital imagery of ancient paintings by mixtures of stochastic models, IEEE Transactions on Image Processing 2004.

[43] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui and M. I. Jordan, Learning the Kernel Matrix with Semidefinite Programming, JMLR 2004.

[44] S. Sonnenburg, G. Rätsch, C. Schäfer and B. Schölkopf, Large Scale Multiple Kernel Learning, JMLR 2006.

[45] R. Xiao, W. Li, Y. Tian and X. Tang, Joint Boosting Feature Selection for Robust Face Recognition, CVPR 2006.

[46] K. Grauman and T. Darrell, The Pyramid Match Kernel: Efficient Learning with Sets of Features, JMLR 2007.

[47] K. Crammer, J. Keshet and Yoram Singer, Kernel Design Using Boosting, NIPS 2002.

[48] T. Hertz, A. Bar Hillel and Daphna Weinshall, Learning a kernel function for classification with small training samples, ICML 2006.

[49] W. Jiang, S. Chang and A. C. Loui, Kernel Sharing With Joint Boosting For Multi-Class Concept Detection, CVPR 2007.

[50] P. Viola, M. J. Jones and D. Snow, Detecting pedestrians using patterns of motion and appearance, ICCV 2003.

[51] Y. Freund and R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, European Conference on Computational Learning Theory 1995.

[52] Y. Ke and R. Sukthankar, PCA-SIFT: a more distinctive representation for local image descriptors, CVPR 2004

[53] M. Varma and A. Zisserman, Texture Classification: Are Filter Banks Necessary?, CVPR 2003.

[54] W. T. Freeman and E. H. Adelson, The Design and Use of Steerable Filters, IEEE Transactions on PAMI 1991.

[55] C. J. C. Burges, Simplified Support Vector Decision Rules, ICML 1996.

[56] J. Friedman, T. Hastie, and R. Tibshirani, Additive Logistic Regression: A Statistical View of Boosting, The Annals of Statistics 2000.

[57] http://research.graphicon.ru/machinelearning/gmladaboostmatlabtoolbox.html/.

[58] http://archive.ics.uci.edu/ml/.

[59] Y. Rubner, C. Tomasi and L. J. Guibas, The Earth Mover's Distance as a Metric for Image Retrieval, IJCV 2000.

[60] A. Bosch, A. Zisserman and X. Munoz, Representing shape with a spatial pyramid kernel, ACM-CIVR 2007.

[61] M. Varma and D. Ray, Learning The Discriminative Power-Invariance Trade-Off, ICCV 2007.

[62] J. Bi, T. Zhang and K. P. Bennett, Column-generation boosting methods for mixture of kernels, ACM-SIGKDD 2004.

[63] R. Arnheim, Art and Visual Perception. A Psychology of the Creative Eye, The University of Chicago Press 1955.

[64] L. Leslie, C. Tat-Seng Chua and Ramesh Jain, Annotation of paintings with high-level semantic concepts using transductive inference and ontology-based concept disambiguation, ACM-MM 2007.

[65] A. Torralba, Contextual priming for object detection, IJCV 2003.

[66] J. Petterson, T. S. Caetano, J. J. McAuley and J. Yue, Exponential Family Graph Matching and Ranking, NIPS 2009.

[67] J. Petterson and T. S. Caetano, Reverse Multi-Label Learning, NIPS 2010.

[68] C. H. Lampert, H. Nickisch and S. Harmeling, Learning To Detect Unseen Object Classes by Between-Class Attribute Transfer, CVPR 2009.

[69] A. Farhadi, I. Endres and D. Hoiem and D. Forsyth, Describing Objects by their Attributes, CVPR 2009.

[70] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn and A. Zisserman, The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results.

[71] Gary B. Huang, Manu Ramesh, Tamara Berg and Erik Learned-Miller, Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments, 2007.

[72] T. Joachims, Optimizing Search Engines Using Clickthrough Data, KDD 2002.

[73] Y. Freund, R. Iyer, R. E. Schapire and Y. Singer, An Efficient Boosting Algorithm for Combining Preferences, JMLR 2003.

[74] Q. V. Le and A. J. Smola, Direct optimization of ranking measures, "http://arxiv.org/abs/0704.3359", 2007.

[75] M. Guillaumin, T. Mensink, J. Verbeek and C. Schmid, Discriminative metric learning in nearest neighbor models for image auto-annotation, ICCV 2009.

[76] H. M. Kuhn, The Hungarian method for the assignement problem, Naval Research Logistics Quarterly, 1955.

[77] N. Kumar, A. C. Berg, P. N. Belhumeur and S. K. Nayar, Attribute and Simile Classifiers for Face Verification, ICCV 2009.

[78] N. Kumar, P. Belhumeur and S. Nayar, FaceTracer: A Search Engine for Large Collections of Images with Faces, ECCV 2008.

[79] Y. Wang and G. Mori, A Discriminative Latent Model of Object Classes and Attributes, ECCV 2010.

[80] D. Grangier and S. Bengio, A Discriminative Kernel-based Model to Rank Images from Text Queries, IEEE PAMI 2008.

[81] T. Berg and D. Forsyth, Animals on the Web, CVPR 2006.

[82] R. Fergus, L. Fei-Fei, P. Perona and A. Zisserman, Learning Object Categories using Google's Image Search, ICCV 2005.

[83] J. Krapac, M. Allan, J. Verbeek and F. Jurie, Improving web image search results using query-relative classifiers, CVPR 2010.

[84] G. Wang and D. Forsyth, Object image retrieval by exploiting online knowledge resources, CVPR 2008.

[85] C. J. C. Burges, R. Ragno and Q. V. Le, Learning to rank with nonsmooth cost functions, NIPS 2006.

[86] Y. Hu and M. Liu and N. Yu, Multiple-instance ranking: learning to rank images for image retrieval, CVPR 2008.

[87] C. Teo, S. Vishwanathan, A. Smola and Q. Le, Bundle methods for regularized risk minimization, JMLR 2010.

[88] D. A. Vaquero, R. S. Feris, Duan Tran, Lisa Brown, Arun Hampapur and M. Turk, Attribute-Based People Search in Surveillance Environments, WACV 2009.

[89] P. Duygulu, K. Barnard, N. de Freitas and D. Forsyth, Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary, ECCV 2002.

[90] O. Barinova, V. Lempitsky, E. Tretyak and P. Kohli, Geometric image parsing in man-made environments, ECCV 2010.

[91] J. P. Tardif, Non-iterative approach for fast and accurate vanishing point detection, ICCV 2009.

[92] K. Saenko, B. Kulis, M. Fritz and T. Darrell, Adapting Visual Category Models to New Domains, ECCV 2010.

[93] J. Yang, R. Yan and A. G. Hauptmann, Cross-domain video concept detection using adaptive svms, ACM Multimedia 2007.

[94] M. Stark, M. Goesele and B. Schiele, A shape-based object class model for knowledge transfer, ICCV 2009.

[95] L. Duan, I. W. Tsang, D. Xu and S. J. Maybank, Domain transfer svm for video concept detection, CVPR 2009.

[96] S. J. Pan and Q. Yang, A Survey on Transfer Learning, IEEE Transactions on Knowledge and Data Engineering, 2010.

[97] D. Hoiem, A. A. Efros and M. Hebert, Putting Objects in Perspective, IJCV 2008.

[98] C. Gu and X. Ren, Discriminative Mixture-of-Templates for Viewpoint Classification, ECCV 2010.

[99] S. Ying-Ze, B. M. Sun and S. Savarese, Toward Coherent Object Detection And Scene Layout Understanding, CVPR 2010.

[100] S. Savarese and L. Fei-Fei, 3D generic object categorization, localization and pose estimation, ICCV 2007.

[101] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele and L. Van Gool, Towards Multi-View Object Class Detection, CVPR 2006.

[102] Y. Yang, J. Liu and M. Shah, Video Scene Understanding with Multi-Scale Analysis, ICCV 2009.

[103] L. Zelnik-Manor and P. Perona, Self-tuning spectral clustering, NIPS 2004.

[104] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, Object detection with discriminatively trained part based models, IEEE PAMI 2010.

[105] A. Torralba, K. Murphy and W. Freeman, Sharing features: efficient boosting procedures for multiclass object detection, CVPR 2004.

[106] P. Yan, S. M. Khan and M. Shah, 3D Model based Object Class Detection in An Arbitrary View, ICCV 2007.

[107] J. Koenderink and A. van Doorn, The internal representation of solid shape with respect to vision, Biological Cybernetics 1979.

[108] V. Hedau, D. Hoiem and D. Forsyth, Thinking Inside the Box: Using Appearance Models and Context Based on Room Geometry, ECCV 2010.

[109] J. Blitzer, M. Dredze and F. Pereira, Biographies, Bollywood, Boom-boxes, and Blenders: Domain Adaptation for Sentiment Classification, ACL 2007.

[110] H. Su, M. Sun, L. Fei-Fei and S. Savarese, Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories, ICCV 2009.

[111] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results,

[112] H. Su, M. Sun, L. Fei-Fei and S. Savarese, Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories, ICCV 2009.

[113] C. Liu and J. Yuen and A. Torralba, Nonparametric Scene Parsing: Label Transfer via Dense Scene Alignment, CVPR 2009.

[114] T. Malisiewicz and A. A. Efros, Beyond Categories: The Visual Memex Model for Reasoning About Object Relationships, NIPS 2009.

[115] O. Boiman, E. Shechtman and M. Irani, In Defense of Nearest-Neighbor Based Image Classification, CVPR 2008.

[116] B. Siddiquie, B. White, X. Chen and L. S. Davis, Multi-view Hashing for Multi-Modal Image Retrieval based on Complex Descriptive Queries.

[117] B. Siddiquie, R. S. Feris and L. S. Davis, Image Ranking and Retrieval based on Multi-Attribute Queries, CVPR 2011.

[118] B. Siddiquie and A. Gupta, Beyond Active Noun Tagging: Modeling Contextual Interactions for Multi-Class Active Learning, CVPR 2010.

[119] B. Siddiquie, R. S. Feris and L. S. Davis, Unsupervised Transfer Learning for View-Invariant Object Recognition.

[120] B. Siddiquie, S. N. Vitaladevuni and L. S. Davis, Combining Multiple Kernels for Efficient Image Classification, WACV 2009.

[121] A. Torralba, R. Fergus and Y. Weiss, Small codes and large databases for recognition, CVPR 2008.

[122] H. Jegou, M. Douze and C. Schmid, Aggregating local descriptors into a compact image representation, CVPR 2010.

[123] F. Perronnin, Y. Liu, J. Sanchez and H. Poirier, Large-Scale Image Retrieval with Compressed Fisher Vectors, CVPR 2010.

[124] M. Perd'och, O. Chum and J. Matas, Effcient Representation of Local Geometry for Large Scale Object Retrieval, CVPR 2009.

[125] C. E. Jacobs, A. Finkelstein and D. H. Salesin, Fast multi resolution image querying, ACM SIGGRAPH 1995.

[126] J. R. Smith and S.-F. Chang, A Fully Automated Content-Based Image Query System, ACM Multimedia 1996.

[127] H. Xu, J. Wang, X.-S. Hua and S. Li, Image Search by Concept Map, SIGIR 2010.

[128] T. Chen, M. Cheng, P. Tan, A. Shamir and S. Hu, Sketch2Photo: Internet Image Montage, SIGGRAPH ASIA 2009.

[129] Y. Cao, W. Changhu, Z. Liqing and L. Zhang, Edgel Inverted Index for Large-Scale Sketch-based Image Search, CVPR 2011.

[130] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C Berg and T. L Berg, Baby Talk: Understanding and Generating Simple Image Descriptions, CVPR 2011.

[131] A. Farhadi, M. Hejrati, A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier and D. Forsyth, Every Picture Tells a Story: Generating Sentences for Images, ECCV 2010.

[132] D. Grangier and S. Bengio, A Discriminative Kernel-based Model to Rank Images from Text Queries, PAMI 2008.

[133] M. Guillaumin, T. Mensink, J. Verbeek and C. Schmid, TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation, ICCV 2009.

[134] M. Datar, N. Immorlica, P. Indyk and V. Mirrokni, Locality-Sensitive Hashing Scheme Based on p-Stable Distributions, SOCG 2004.

[135] B. Kulis and K. Grauman, Kernelized locality-sensitive hashing for scalable image search, ICCV 2009.

[136] R. Salakhutdinov and G. Hinton, Semantic hashing, SIGIR 2007.

[137] J. Wang, S. Kumar and S.-F. Chang, Semi-supervised hashing for large-scale image retrieval, CVPR 2010.

[138] M. Raginsky and S. Lazebnik, Locality sensitive binary codes from shift-invariant kernels, NIPS 2009.

[139] Spectral hashing, Y. Weiss, A. Torralba and R. Fergus, NIPS 2008.

[140] Y. Gong and S. Lazebnik, Iterative Quantization: A Procrustean Approach to Learning Binary Codes, CVPR 2011.

[141] L.-J. Li, H. Su, E. Xing and L. Fei-Fei, Object Bank: A High-Level Image Representation for Scene Classification and Semantic Feature Sparsification, NIPS 2010.

[142] Y. Chen and Y. Weiss, Finding the M Most Probable Configurations Using Loopy Belief Propagation, NIPS 2004.

[143] D. Park and D. Ramanan, N-best maximal decoders for part models, ICCV 2011.

[144] Y. Zhang, Z. Jia and T. Chen, Image Retrieval with Geometry-Preserving Visual Phrases, CVPR 2011.

[145] D. Radev, W. Fan, H. Qi, H. Wu and A. Grewal, Probabilistic question answering on the Web, Journal of the American Society for Information Science and Technology, 2005.

[146] A. Sharma and D. W. Jacobs, Bypassing Synthesis: PLS for Face Recognition with Pose, Low-Resolution and Sketch, CVPR 2011.

[147] D. Parikh and K. Grauman, Relative Attributes, ICCV 2011.

[148] J. Shotton, M. Johnson and R. Cipolla, Semantic Texton Forests for Image Categorization and Segmentation, CVPR 2008.